A PHOTOGRAMMETRIC ON-ORBIT INSPECTION FOR ORBITER THERMAL PROTECTION SYSTEM

A Thesis

by

PETER PAUL GESTING

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

December 2005

Major Subject: Aerospace Engineering

A PHOTOGRAMMETRIC ON-ORBIT INSPECTION FOR ORBITER THERMAL PROTECTION SYSTEM

A Thesis

by

PETER PAUL GESTING

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	John E Hurtado
Committee Members,	John Junkins
	John Valasek
	Reza Langari
Head of Department,	Helen Reed

December 2005

Major Subject: Aerospace Engineering

ABSTRACT

A Photogrammetric On-Orbit Inspection for
Orbiter Thermal Protection System. (December 2005)
Peter Paul Gesting, B.S., Texas A&M University
Chair of Advisory Committee: Dr. John E. Hurtado

Due to the Columbia Space Shuttle Accident of February 2003, the Columbia Accident Investigation Board determined the need for an on-orbit inspection system for the Thermal Protection System that accurately determines damage depth to 0.25". NASA contracted the Spacecraft Technology Center in College Station, Texas, for a proof-of-concept photogrammetric system. This system involves a high quality digital camera placed on the International Space Station, capable of taking high fidelity images of the orbiter as it rotates through the Rendezvous Pitch Maneuver. Due to the pitch rotation, the images are tilted at different angles. The tilt causes the damage to exhibit parallax between multiple images. The tilted images are therefore registered to the near-vertical images using visually striking features on the undamaged surface of the Thermal Protection System that appear in multiple images taken at different tilt angles. The images become relatively oriented after registration, and features in one image are ensured to lie on the epipolar line in the other images. Features that do not lie on the undamaged surface, however, are shifted in the tilted images. These pixels are matched to the near-vertical image using a sliding-window area-matching approach. The windows are matched using a least-squares error method. The change in location for a pixel in a tilted image from its expected location on the undamaged surface is called the pixel disparity. This disparity is linearly scaled using the tilt angle and the pixel sampling to determine the depth of the damage at that pixel

location. The algorithm is tested on a set of damaged tiles at the Johnson Space Center in Houston and the photogrammetric damage depth is then compared to a set of truth data provided by NASA. The photogrammetric method shows promise, with the 0.25" error limit being exceeded in only a few pixel locations. Once the camera properties are fully known from calibration, this systematic error should be reduced.

Now therefore and always be ascribed as is most justly due all might, dominion, majesty, glory, honor, and power to the Unity indivisible, the Trinity of sovereign love, God the Father, Son, and Holy Spirit.

ὅτι ἐξ αὐτοῦ καὶ δἰ αὐτοῦ καὶ εἰς αὐτὸν τὰ πάντα· αὐτῷ ἡ δόξα εἰς τοὺς αἰῶνας, ἀμήν.

-ΠΡΟΣ ΡΩΜΑΙΟΥΣ

ACKNOWLEDGMENTS

What did I do?

Did I choose to be born at a time in history when higher education is so readily available? Did I choose to be born at such a time in a country where higher education is so highly valued and encouraged? Did I choose to be born into such an amazing family, with parents that did nothing but nurture and instill in me a desire for higher education?

Was it because of an acquired wisdom that I was surrounded by such a host of friends who encouraged and edified me? Was it my choice to be blessed with a great adviser like Dr. John Hurtado, or a great project manager like Diane Hurtado? Or was it my choice that Dr. Mark Lemmon took the time and effort to educate me in his research?

Could I choose such blessings? Could I choose to be honored with an ability to learn and be taught? Can I take credit for the gifts and talents bestowed upon me in the infinite wisdom, grace, and mercy of a Holy God? Can I owe nothing but praise and adoration for the blessings bestowed and the people provided? Could I have followed any other course than that which was set before me? For as Christ Jesus stated,

> Which of you, having a slave plowing or tending sheep, will say to him when he has come in from the field, 'Come immediately and sit down to eat'? But will he not say to him, 'Prepare something for me to eat, and properly clothe yourself and serve me while I eat and drink; and afterward you may eat and drink'? He does not thank the slave because he did the things which were commanded, does he? So you too, when you do all the things which are commanded you, say, 'We are unworthy slaves; we have

done only that which we ought to have done.'

-Luke 17:7-10

What did I do? I showed up.

TABLE OF CONTENTS

CHAPTER Page

LIST OF FIGURES

FIGURE Page

FIGURE Page

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION AND BACKGROUND

The space shuttle Columbia disaster of February 2003 brought to the forefront of NASA's attention the danger posed to the space shuttle tile Thermal Protection System (TPS). The thermal tile system is designed to ensure that the enormous heat generated upon orbiter re-entry does not harm the orbiter or its crew. Upon inspection, the Columbia Accident Investigation Board (CAIB) determined that the cause of the crash was a piece of foam from the external tank that broke off during ascent and impacted the leading edge of Columbia's left wing, damaging the Reinforced Carbon-Carbon (RCC) panels. Upon re-entry, hot gases of up to 3000°F entered the breach and caused the break-up of Columbia over the Western United States[1].

The Board mandated three major improvements that were necessary before a return to flight and required as part of all future flights. These improvements were a redesign of critical points of the external tank foam that were likely to break off, an increase in the visual monitoring of the orbiter during take-off and ascent, and an on-orbit inspection of the tile TPS[2]. NASA decided that an on-orbit thermal tile inspection system would have to be implemented for the remainder of the shuttle missions. NASA is currently looking at multiple tile inspection systems, one of which has already flown[3, 4].

The most common cause of damage to the orbiter thermal tiles is from the foam placed on the external tank to keep the liquid hydrogen and oxygen fuel cold. During takeoff and ascent through the atmosphere, the force of the engines causes a great deal of vibration, which can cause the foam to shed off the tank. The piece that shed

---

The journal model is *IEEE Transactions on Automatic Control.*

off Columbia's external tank was from the bipod ramp, where the ET is attached to the orbiter. Due to the speed of the orbiter through ascent and the drag associated while still in the atmosphere, pieces of foam can be accelerated to very high speeds after being shed; in the case of the Columbia accident, up to 500 mph[1]. Therefore in addition to an on-orbit tile inspection system, NASA has also redesigned the ET foam so as to reduce the chance of large chunks of foam falling off and damaging the tiles[5].

A. Inspection Requirements

NASA has contracted the Spacecraft Technology Center (STC) to provide a means of detecting damage to the orbiter thermal tile protection system while on-orbit. The current active system using the robotic arm laser/camera system is bulky, time consuming, and expensive. It also requires a 50ft extension to the orbiter boom arm. Because this system is an active laser system, the tip of the boom and extension must be maneuvered beneath the orbiter to image the bottom of the vehicle. There is therefore an increased chance for damage to the TPS tiles from inadvertently impacting the tiles with the boom. This requires significant time and training for the boom operator to ensure a safe operating procedure. Therefore a significant portion of mission time must be used for the boom imaging.

The requirements of the inspection system are to be able to resolve damage depth correctly to within 0.25" and resolve damaged areas laterally as small as 0.3". The STC, along with researchers from Florida Atlantic University (FAU) and Texas A&M University, has proposed a passive system using a high resolution camera stationed on the International Space Station (ISS). The researchers at FAU have developed a high resolution black and white digital camera for this project. The camera will be based

on the ISS and positioned to look in the nadir direction toward Earth. During the Rendezvous Pitch Maneuver (RPM), which the orbiter will perform before docking with the ISS, the camera will take pictures of the bottom of the orbiter as it spins 600ft beneath the ISS. Using a 600mm lens and a 1.4x optical extender with a pixel base of 3840x2160, the camera will be able to take high fidelity images of the bottom of the orbiter. Since the images are in black and white, only gray levels will be available in the images. An example of the image is given in Figure 1. This image is a simulation of the fidelity that will be obtained from this system.

B. Research Objectives

Photogrammetry has been primarily used for aerial photos[6, 7, 8]. Aerial photogrammetry is primarily used for certain feature points that are available in two images to determine an altitude map. Machine vision is a more modern approach that uses close range photogrammetry and a few distinct points to determine surface geometry[9]. This research will use certain aspects of both approaches, for greater precision than aerial photogrammetry and at a greater distance than machine vision.

For aerial photogrammetry, most of the development has been for vertical or near-vertical photographs, where the camera and the plane of the object space are perpendicular. Moving the camera from one position to another while keeping the camera perpendicular to the object space can induce parallax for objects that are different distances from the camera. There are several factors, however, that prevent this setup for our current problem. First, a requirement from NASA is that the inspection system be able to see lateral features on the tile at a resolution of 0.3". Given the size of the orbiter (120'x80'), the size of the pixel base would have to be enormous to accommodate the same features in a vertical alignment and a lateral shifting of the

Fig. 1. Example Tile Image.

orbiter while still resolving features of this size. Second, it is not practically feasible to maneuver the orbiter in an accommodating manner: in order for the bottom of the orbiter to be exposed to the camera, the orbiter must be 'upside down' with respect to the ISS, so the orbiter pilot will be unable to maintain visual reference with the ISS. It would therefore be extremely dangerous in close relative motion maneuvers without visual reference to try to shift the orbiter laterally. However, photogrammetry still requires parallax to determine the depth of the damage. Therefore the camera will

take images as the orbiter pitches through a full rotation during the RPM. Instead of moving the camera with a stationary object, parallax is introduced by rotating the object through many angles before a stationary camera. Therefore a further study into the depth due to parallax from non-vertical images will be undertaken to satisfy this physical constraint.

The scope of this research is to undertake to develop a set of photogrammetry algorithms to determine the damage depth from the high resolution photos. Given a set of images from the high resolution camera of a damaged area, these algorithms should correctly resolve the lateral and depth dimensions of damaged areas to within the required specifications. Because the images will not be vertical images with parallax from changing camera position, the algorithms will need to take into account the orientation and position of the tiles in the images and determine the respective parallax.

The default sampling will be the field of view of the camera with the lens and extender divided by the pixel base. If this sampling is too small, then the pixel to pixel noise will give false information. If the sampling is too large, however, then the necessary resolution from the project specifications will not be met. There is an inverse correlation of the resolution of the lateral features to the depth of the damage, therefore the resolution will have to be great enough to not alias the photos, but not so great that the noise in one direction requires a re-sampling with the results of losing resolution in the other.

CHAPTER II

PHOTOGRAMMETRY

The idea of photogrammetry has been around for many decades. This concept is able to determine relative and absolute positions and orientations of objects from photographs, based upon knowledge of the camera properties[10]. Much of the original and current use has been for aerial photography and elevation data.

A. Parallax

The difference in perspective between two objects in two different photos based upon change in the position of either the object or camera is called parallax. Using two pictures of the same feature taken from different camera positions, one can determine relative and absolute elevation (relative to a datum) from the parallax between the two photos, as seen in Figure 2, where P is the focal plane, B is the photo base, f is the focal length, and the parallax is given by $x_1 - x_2$. Taking the triangle (B_2, O_2, a_2)

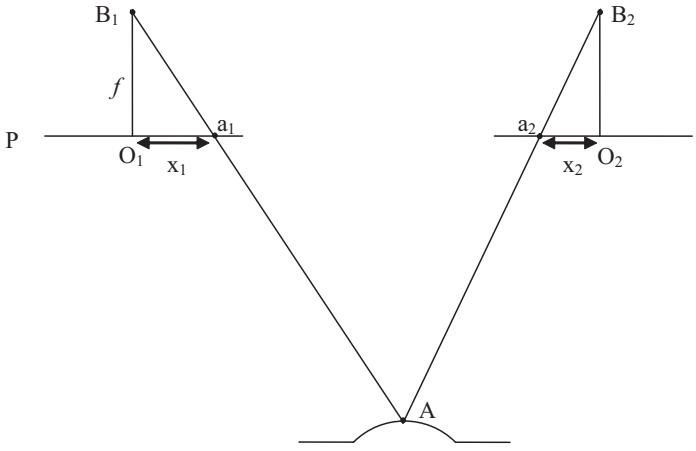

Fig. 2. Example of Parallax.

and placing it alongside triangle (B_1, O_1, a_1) produces the drawing in Figure 3. Using

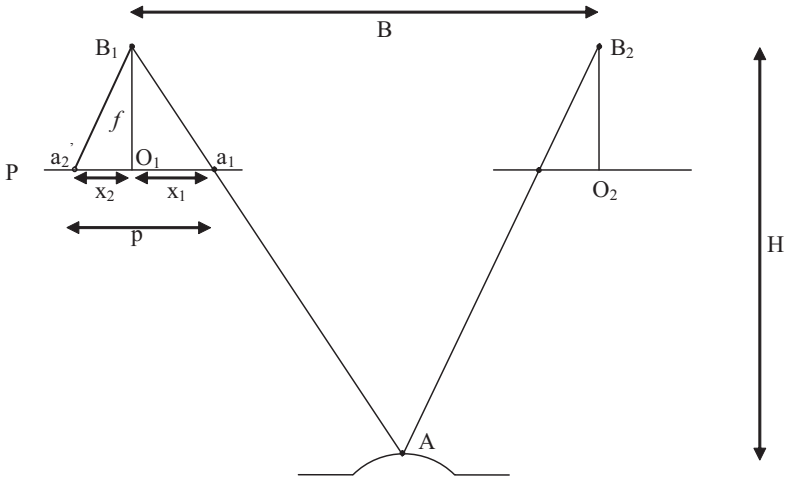

Fig. 3. Similar Triangles.

similar triangles (B_1, B_2, A) and (B_1, a_1, a_2') one can obtain Equation 2.1, where H is the elevation of point A.

$$\frac{B}{H} = \frac{p}{f} \tag{2.1}$$

In our application, however, we will be interested in the change of elevation from two different object points. For object point 1 with elevation h_1 and parallax p_1, and object point 2 with elevation h_2 and parallax p_2, one can show Equation 2.2.

$$\begin{aligned} \Delta h &= h_2 - h_1 = \frac{Bf}{p_2} - \frac{Bf}{p_1} \\ \Delta h &= Bf\left(\frac{1}{p_2} - \frac{1}{p_1}\right) = Bf\left(\frac{\Delta p}{p_1 p_2}\right) \end{aligned} \tag{2.2}$$

This formulation has made use of a number of assumptions, including equal exposure height, equal focal length, and equal and vertical exposure stations. However, it is the basic formulation for the project[11].

CHAPTER III

DIGITAL PHOTOGRAMMETRY

Photogrammetry was developed for, and has traditionally been used for, analog, or film images. However, the projective mathematical concepts that were developed for analog images are the same when applied to digital images. Digital images, however, provide many advantages to analog images. Digital images are stored in a computer, therefore a computer can be used to perform specific processes uniformly to the image very quickly. Also, a computer can be programmed to automate tasks that would normally be performed by a human.

A. Image Sampling

Digital images sample a continuous analog image at discrete intervals. It is therefore important to determine whether all the information from the analog signal (or image) is captured in the digital sampling. Detailed derivations of sampling theory can be found in standard references such as Bracewell[12] and Oppenheim and Schafer[13].

The sampling rate required to reconstruct an analog signal depends on how fast the signal varies, which is called the frequency. This is based upon the Fourier transform, where a combination of sine and cosine functions are used to represent an analog signal. The faster a signal varies, therefore, the higher the frequency of the sine and cosine functions in the Fourier transform required to model the signal accurately[11]. Sampling theory titles the minimum sampling required to exactly reconstruct a signal the Nyquist rate. To reconstruct a signal with a highest frequency component ω, a sampling rate of 2ω is required[11].

Slowing varying features in images have low frequency components and can be reconstructed with less sampling points. Fast varying features, such as edges, have

high frequency components and require more sampling points to reconstruct.

If a signal is not sampled fast enough, the high frequency components are incorrectly interpreted as low frequency components. This causes a loss of information, known as aliasing. Spatial aliasing in images usually involves the loss of sharp edges in digital images that are sampled incorrectly.

Digital images are also quantized samples of the analog signal. These values are based upon the amount of light that is incident upon the digital sensor. Noise, therefore, is inherent in digital images through this quantization. The amount of noise from this quantization depends on the quantization levels. Typical 8-bit images have gray values that range from 0 to 255. Because similar values between quantization levels are given the same value, the amount of noise can be reduced by choosing quantized values that are based upon the probability distribution of the image data[11]. In the current project, this is handled through the camera by balancing the white values so that a distributed histogram of values is obtained.

B. Image Resampling

Due to the image sampling difficulties of the previous section, and in addition to other concerns, it usually becomes necessary to resample a digital image. There are many reasons to resample, but the primary one pertains to image registration, which will be discussed in Chapter IV Section C. Because digital images are stored on a computer, automating the resampling is simple and efficient. The biggest issue in resampling is the same one that occurs in sampling — the problem of aliasing. If resampling at a lower rate, precautions must be taken to prevent aliasing. However, because resampling in this project will not change the number of sample points in the image, nor the number of points from which to sample (i.e. the sampling frequency

will be the same in resampling), aliasing will not be a problem in resampling.

The need for resampling within this research will arise from two processes. The first deals with the fact that the camera is not taking vertical images. Because the images are tilted, they must first be adjusted to make them the same scale as vertical images. Next, the images must be registered to the vertical images, which is explained in Chapter IV Section C. These processes will create new pixel locations for the same size image. Therefore, values at fractional pixel locations will need to be determined.

1. Bilinear Interpolation

There are many ways which to resample. Nearest neighbor interpolation uses the closest pixel to the transformed coordinate. This is the least expensive and quickest resampling method. Bilinear interpolation is a much more accurate and more widely used approach. Given a transformed coordinate w_0 at (x_0,y_0) that lies between coordinates (x_i,y_j), (x_{i+1},y_j), (x_i,y_{j+1}), (x_{i+1},y_{j+1}), as shown in Figure 4, first calculate the auxiliary values t and u as shown in Equations 3.1 and 3.2. Then the desired value w_0 can be calculated from Equation 3.3. [11]

$$t = \frac{x_0 - x_i}{x_{i+1} - x_i} \tag{3.1}$$

$$u = \frac{y_0 - y_j}{y_{j+1} - y_j} \tag{3.2}$$

$$w_0 = (1-t)(1-u)w_{i,j} + t(1-u)w_{i+1,j} + tuw_{i+1,j+1} + (1-t)uw_{i,j+1} \tag{3.3}$$

It can be seen from Equations 3.1 - 3.3 that as the name implies, bilinear interpolation simply assumes a linear variation in both directions between sampling points.

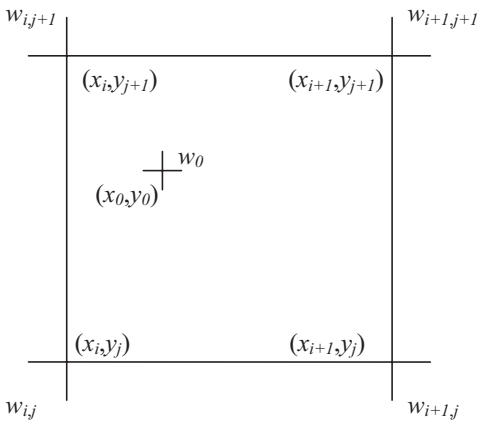

Fig. 4. Bilinear Interpolation.

2. Bicubic Interpolation

Another method for resampling is known as bicubic interpolation. Assuming that the image was sampled above the Nyquist rate, or sampled at least twice as high as the highest spatial frequency desired, the image can be exactly reconstructed using the *sinc* function, as shown in Equation 3.4.

$$sinc\,(x) = \frac{sin\,(\pi x)}{\pi x} \tag{3.4}$$

Because the *sinc* function is based upon the *sin* function, it is affected by an infinite number of pixel values far from the center. However, due to the πx in the denominator, the influence of each successive pixel is reduced the farther it is from the point in question. Therefore this function is usually approximated by only considering a few pixel locations near the fractional pixel in question. The most common approach is to approximate the *sinc* function with a cubic spline. The equations for this

approximation are given in Equations 3.5 - 3.7

$$f_1(x) = (a+2)x^3 - (a+3)x^2 + 1 \qquad \text{for } 0 \le x \le 1 \tag{3.5}$$

$$f_2(x) = ax^3 - 5ax^2 + 8ax - 4a \qquad \text{for } 1 \le x \le 2 \tag{3.6}$$

$$f_3(x) = 0 \qquad \text{for } x \ge 2 \tag{3.7}$$

where a is a free parameter used to define the slope of the function at $x = 1$ (usually $a = -0.5$ gives good results), and x is the absolute value of the difference between the whole number pixel row or column and the transformed fractional pixel row or column[10].

It can be seen from Equation 3.7 that this formulation of the bicubic interpolation allows the influence of points at two pixels away and less. This allows for a more accurate interpolation than the bilinear approach, which only allows for the influence of points one pixel away.

With the previous three interpolation schemes, there is a trade-off between accuracy and computational speed. Nearest neighbor interpolation is obviously the computationally least expensive, but also the least accurate. Bicubic interpolation, on the other hand, is the most accurate but more computationally intensive.

C. Edge Enhancement

Many features in an image give rise to perceived edges. These can be depth discontinuities, changes in color or lighting, and changes in surface orientation. It can therefore be beneficial to locate and enhance these edges. For the current research, edges occur at tile edges, serial number markings, and damage boundaries. These edges play an important part in image registration, as will be shown in Chapter IV Section C.

Many methods exist for edge detection, and some of the easiest are based upon gradient operators. Because edges are defined by changes in image intensity, these operators are a natural choice. The easiest gradient operator is known as the Roberts operator. This operator takes in an image and returns an image of the same size where the pixel located at (i,j), with a value of $F_{i,j}$, is replaced by $G_{i,j}$ as in Equation 3.8

$$G_{i,j} = |F_{i,j} - F_{i+1,j+1}| + |F_{i,j+1} - F_{i+1,j}| \tag{3.8}$$

This is equivalent to a convolution using the kernels given as

$$\begin{bmatrix} 0 & -1 \\ \underline{1} & 0 \end{bmatrix} \text{ and } \begin{bmatrix} 1 & 0 \\ \underline{0} & -1 \end{bmatrix}$$

where the underlined values represent the pixel located at (i, j). Typically the last column and row are given a value of zero.

CHAPTER IV

ALGORITHM DEVELOPMENT

The algorithms necessary for the current research will differ from the algorithms developed in Chapter II because the assumption made in Section A of equal and vertical exposure stations does not hold true. The requirements given in Chapter I Section B showed that it is practically infeasible to take multiple vertical images from separate exposure stations. Therefore, it was shown that instead multiple images would be obtained by taking *tilted* images from the same exposure station as the orbiter rotates through the RPM at a constant distance from the camera.

A. Tilted Images

Titled images introduce a number of difficulties that vertical images lack. The main complication comes from the now non-constant scale factor that is introduced. In a vertical image, variation on object distances were caused by topographical elevation. In tilted photographs, however, changes in relief are now caused by a change in scale from one part of the image to another. To use a tilted image for photogrammetric purposes, it is possible to just work with the varying image scale. A method that is chosen for this research, however, will be rectification, which will alter the tilted image to give it better properties for photogrammetric uses.

B. Image Rectification

The process of rectification takes a tilted image and transforms it into an equivalent vertical image. The geometry for the tilted and rectified image is given in Figure 5. In Figure 5, L is the photo exposure station point, LO is a ray from the exposure

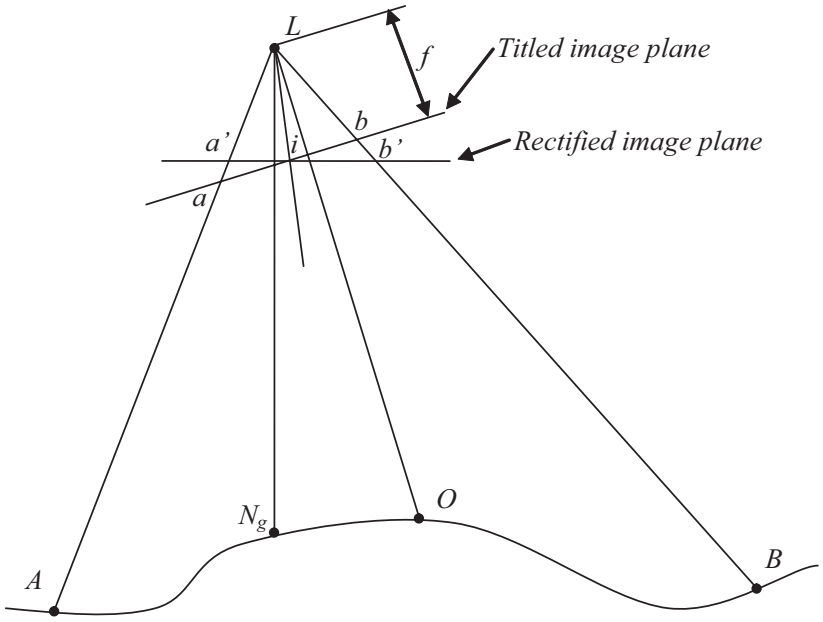

Fig. 5. Geometry of Tilted and Rectified Image.

station perpendicular to the tilted image plane, LN is a ray from the exposure station vertically to the ground, A and B are object points, a and b are the image points in the tilted photo plane, and a' and b' are the image points for object points A and B in the rectified image plane. A rectified image is an equivalent vertical image constructed on an image plane that passes through the *isocenter* of the tilted image. The isocenter of a tilted image is where a ray from the exposure station along the bisector of the tilt angle intersects the titled image plane[10]. In Figure 5 this is point i.

The rectified image is now a vertical image, with the same focal length and properties as if the image was taken as a vertical image in the beginning. Therefore, the photogrammetric methods shown in Chapter II Section A can be used on these rectified images.

Image rectification, like much of analytical photogrammetry, is based upon the *collinearity* conditions that arise from photography geometry. The formulation is complicated, and can be found in Appendix C of Wolf[10].

The result of the rectification formulation is a pair of equations given in Equations 4.1 and 4.2.

$$X = \frac{a_1 x + b_1 y + c_1}{a_3 x + b_3 y + 1} \tag{4.1}$$

$$Y = \frac{a_2 x + b_2 y + c_2}{a_3 x + b_3 y + 1} \tag{4.2}$$

The values X and Y are the ground coordinates of a ground control point, x and y are the image coordinates, and the a_is, b_is, and c_is are eight parameters that describe the orientation of the tilted image with respect to the vertical image. Each ground control point produces a pair of equations in this form. This formulation therefore requires at least four ground control points to solve for the eight unknown orientation parameters. Therefore the object coordinates for the ground control points must be known. However, if more control points are known, then a least squares approach is used to determine the *most likely* values for the eight unknowns.

An extra approach will be used in this research to obtain a better rectified image. Instead of just including image points to the first order, image points to the second order will be used, as shown in Equations 4.3 and 4.4.

$$X = \frac{a_1 x^2 + b_1 y^2 + c_1 x + d_1 y + e_1}{a_3 x^2 + b_3 y^2 + c_3 x + d_3 y + 1} \tag{4.3}$$

$$Y = \frac{a_2 x^2 + b_2 y^2 + c_2 x + d_2 y + e_2}{a_3 x^2 + b_3 y^2 + c_3 x + d_3 y + 1} \tag{4.4}$$

This form is different because it maps the pixel locations to the second order, giving a better approximation than a simple first order mapping. The main reason for this addition is the fact that the systematic errors from the camera are not known. There-

fore instead of the simple translation and rotation that is available from Equations 4.1 and 4.2, Equations 4.3 and 4.4 also allow warping of the undamaged plane. This will help to correct for geometric image distortions, like barreling, that occur. There are now 14 unknowns, and therefore at least seven ground control points are necessary to form a unique solution.

C. Image Registration

Image rectification requires the knowledge of the object coordinates of ground control points. Due to the nature of this problem, however, these object points will not be known. Therefore a different approach must be used. As the shuttle rotates through the RPM, its six-orientation parameters will be recorded and stored. A time stamp will also be recorded on each image. Therefore, for each image that is taken, the orientation of the shuttle, i.e. the tilt angle of the image, will be known. As will be shown in Chapter V Section D and based upon the formulation in Chapter II Section A, the parallax for these images will be determined from one image that is near vertical, and side images that are rectified images from tilted images. Using the knowledge of the tilt angles, it can be determined which images of a certain object point are near vertical and which are tilted. Because object control coordinates are not known for points on the bottom of the orbiter, the tilted images will be rectified to the near vertical image.

The process of rectifying the tilted images taken by the camera and making them useful for the photogrammetric processes shown is called *registration*. A sample image of orbiter tile (a cropped version of Figure 1) is given in Figure 6. The square tiles in Figure 6 are about 4 x 4 inches. This gives a visual idea of the capable resolution. From this image, one can see the damaged areas as the mis-shaped white

Fig. 6. Example Tile Image.

areas. The black area is undamaged tile. One can also see the serial number of each tile written in white. One can see that the undamaged area forms a plane across the black surface with the damaged area receding below it. Therefore this plane in the near vertical image is what will need to be registered from the tilted image. One will notice that the serial numbers on the tiles are easily recognizable and lie in the zero plane of the tile. Therefore these serial numbers will be used as ground control points for the registration.

However, it will not be the object space coordinates that are used as X and Y in Equations 4.3 and 4.4. Instead, the image coordinates of control points in the vertical image will be used to register the serial numbers of the tilted image to that of the vertical image.

1. Epipolar Lines

A pair of images that image the same object point are depicted in Figure 7. This pair of images, with exposure points L_1 and L_2, have an overlapping segment that includes

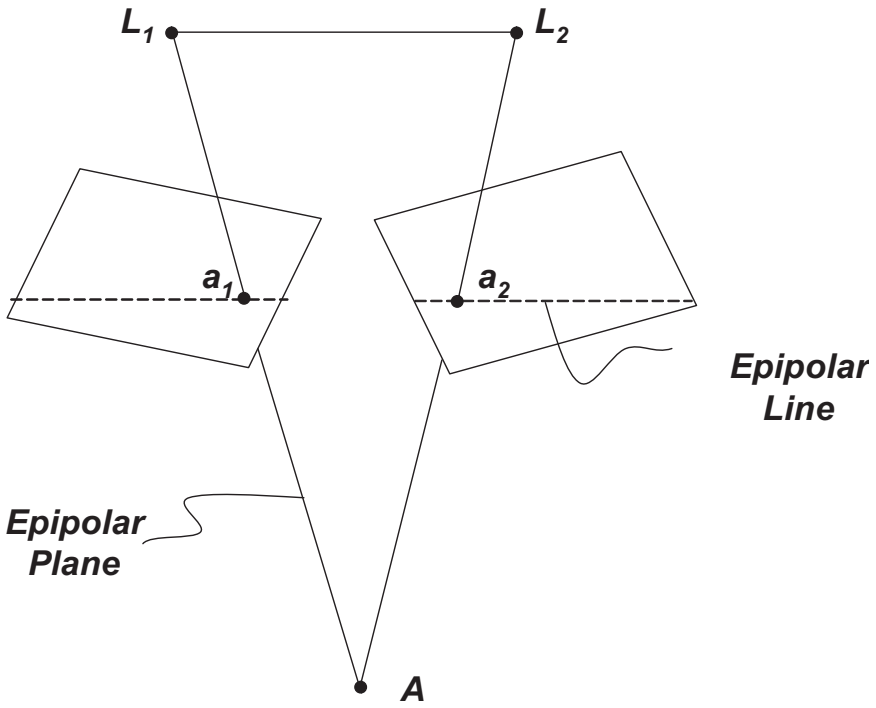

Fig. 7. Epipolar Plane and Lines.

object point A, with image coordinates a_1 and a_2. The plane defined by the points L_1, L_2, and A is called the *epipolar plane*. The lines where this plane intersects each of the images are called *epipolar lines*. These lines provide a very useful service in digital photogrammetry. Once the images are *relatively oriented*, the epipolar lines can be found. Once an object has been found in one image, it can be guaranteed that the object will lie on the epipolar line in the other image.

The image pair must first be relatively oriented to each other so that the epipolar lines correspond in both images. This will be accomplished through the rectification process. As was explained before, the tilted images will be rectified to the near-vertical image with overlapping object points. This process will be done by visually finding points to rectify. Visually striking features in both images will be used to

match the zero plane of the tilted image to the zero plane of the near-vertical image. Then the tilted image will be resampled so that the epipolar lines in the tilted image are parallel to the epipolar lines in the near-vertical image. Once the images are relatively oriented, then when an object is found in one image, it is known that the same object will lie on the epipolar line in the corresponding image. This will prove to be very useful for the next section.

D. Stereo Mapping

Stereo Mapping is the heart of photogrammetry. In order to determine the parallax between two different images of a given object, the image points in both the images must first be found. For analog photos, this is done by aligning the images visually with a *stereocomparator*. A human would align the two images visually, using instrumentation to take into account tilt, scale, and other factors. Then object points in both images would have to be found and the distance from the fiducial center would be found. In analog photogrammetry, the first stage of aligning two images correctly is very tedious and time consuming. However, once that is accomplished, the human eye and brain can very quickly find objects that correspond in both images. In digital photogrammetry, the opposite is true. A computer can be automated, with appropriate input information, to take care of the aligning (through resampling) very quickly. However, finding a feature in one image that corresponds to the same object in another image is very difficult to program into a computer.

Stereo Mapping is intrinsically a difficult problem. Finding a feature that occurs in one image in a corresponding pair image can prove arduous because there will almost always exist an ambiguous match that is not correct. Also, knowing where and how to search can be difficult. However, much of this is alleviated by relatively

orienting the images. If the images are not relatively oriented, then finding a matching feature in an image pair requires a two-dimensional search in one of the images. This can prove to be both time and computationally expensive. However, if the images are relatively oriented, then a feature that is in one image is known to exist on the epipolar line in the other image. Therefore, the search can now be reduced to a one-dimensional search along the epipolar line. Therefore much of the algorithm development for this research was used to correctly register the images together, since this will ease the computational requirements for the matching.

There are two ways to do image matching, known as *feature* based and *area* based matching.

1. Feature-based Matching

Feature matching provides better results than area matching in highly-discontinuous images. Feature matching uses feature points, or contours, to match from one image to another. It is used with good results when looking at aerial photos of urban areas, where building edges are very distinct. It can be used in conjunction with an edge detection algorithm, to find the intersection of an edge-line and an epipolar line. However this can prove to be a problem if the edge is not continuous across the epipolar line in both images. Therefore known constraints must be applied for this method to ensure a correct match.

2. Area-based Matching

Area based matching compares gray-levels of a large area from one image to the next. A window of a certain size centered around the pixel in question is chosen in the master image, and compared to a window of the same size in the slave image. The size of the window is important in this process. A small window is computationally

less expensive, but is highly susceptible to noise. A large window is more resistant to noise, however is more computationally expensive. A larger window is also less likely to find a false positive, because it is trying to match a larger area in both images. Usually what is known as the *sliding window* approach is used, in that the window is shifted along the epipolar line of the slave image and compared to the window of the master image. The distance that the window is shifted in the slave image is based upon the expected disparity between the two images. There are a few ways to compare the two windows to find the best match.

a. Normalized Correlation

Normalized correlation takes into account both the brightness levels and the variance between the two images. The correlation is shown in Equation 4.5.

$$N = \frac{E\left(w_1 w_2\right) - E\left(w_1\right) E\left(w_2\right)}{\sigma\left(w_1\right) \sigma\left(w_2\right)} \tag{4.5}$$

The normalized correlation takes into account the expected values of the pixel values in each of the windows, $E\left(w\right)$, the expected values of the product of the pixel values of the two windows, $E\left(w_1 w_2\right)$, as well as the standard deviations of the pixel values in each of the images, $\sigma\left(w\right)$. This method is very effective when looking at images that have varying brightness levels, since this method normalizes the pixel values to the expected value of the window[11].

b. Least Squares Correlation

If the images have comparable brightness levels, then a least squares approach can be used. The advantage of using a least squares approach is that certain pixel values in the window can be given a higher consideration than others. This is advantageous because a larger window can be used, thereby reducing the impact of noise, but it

also can set the parameters for the area that is to be matched. This is done by using a weight matrix, W, of the same size as the window used. Usually, the pixel at the center of the window is the pixel to be matched, therefore it should be given the highest precedence. As you move away from this pixel, the importance of each successive pixel decreases. Therefore an exponentially decrease weight matrix is used, where each successive pixel is reduced in importance. An example of such a weight matrix for a 5x7 pixel window is shown in Equation 4.6, where each successive pixel is reduced by a factor of 2.

$$W = \begin{bmatrix} 0.03125 & 0.0625 & 0.125 & 0.25 & 0.125 & 0.0625 & 0.03125 \\ 0.0625 & 0.125 & 0.25 & 0.5 & 0.25 & 0.125 & 0.0625 \\ 0.125 & 0.25 & 0.5 & 1 & 0.5 & 0.25 & 0.125 \\ 0.0625 & 0.125 & 0.25 & 0.5 & 0.25 & 0.125 & 0.0625 \\ 0.03125 & 0.0625 & 0.125 & 0.25 & 0.125 & 0.0625 & 0.03125 \end{bmatrix} \tag{4.6}$$

Equation 4.7 gives the expression for the least squares error, E, that is to be minimized, where $w_{1,ij}$ is the pixel value in image 1 at pixel location i, j.

$$E = \sum_i \sum_j W_{ij} \left(w_{1,ij} - w_{2,ij} \right)^2 \tag{4.7}$$

Due to the summation over all the values in the window, to keep everything to scale, the weight matrix is normalized by the sum of all its values so as not to interfere with the final answer.

Figure 6 shows the texture of the damage that is to be matched. One can easily see that there are no distinct features in the damaged areas, other than the edges of the damage itself. Also, each pixel in the damaged area needs to be matched to obtain the required resolution. Therefore, feature based matching is not a viable option for this research. Due to the potential differences in lighting and the ability

to weight values in the window differently, an area-based matching using the least squares correlation is used.

E. Disparity

Once the images are relatively oriented to each other, a few striking features can be found visually in both images and a basic mapping can be found such that a pixel in one image can be transformed to give the corresponding pixel in the other image. This can provide a starting point for the moving window approach outlined above. This mapping is usually performed on ground control points. For this research, it will based upon the zero-plane of the near-vertical image, which has features that are used for registration. Given this parameter map and a starting point for the moving window approach, pixels that lie in the damaged areas of the near-vertical (master) image can be area matched to their corresponding feature in the tilted (slave) image. The parameter map gives a starting point for the matching, but it also provides a baseline for where the damaged pixel would lie in the slave image if the damaged feature lay on the zero/undamaged plane. However, since the feature *is* damaged, it will not be at this position. Figure 8 shows this lateral shifting, or *disparity*, due to depth damage. In Figure 8, point A is a point in the damaged area and A' is where point A would be if there was no damage. L_1 is the exposure station of the vertical image, and L_2 is the exposure station of the rectified tilted image. As can be seen in the vertical image, both points a_1 and a_1' lie in the same position in the image plane of the vertical image. In the rectified tilted image, however, the image points a_2 and a_2' lie in different positions in the image plane. After the parametric mapping, which maps points on the zero-plane non-damaged surface from the vertical image to the rectified tilted image, point a_1 in the vertical image, which corresponds to object

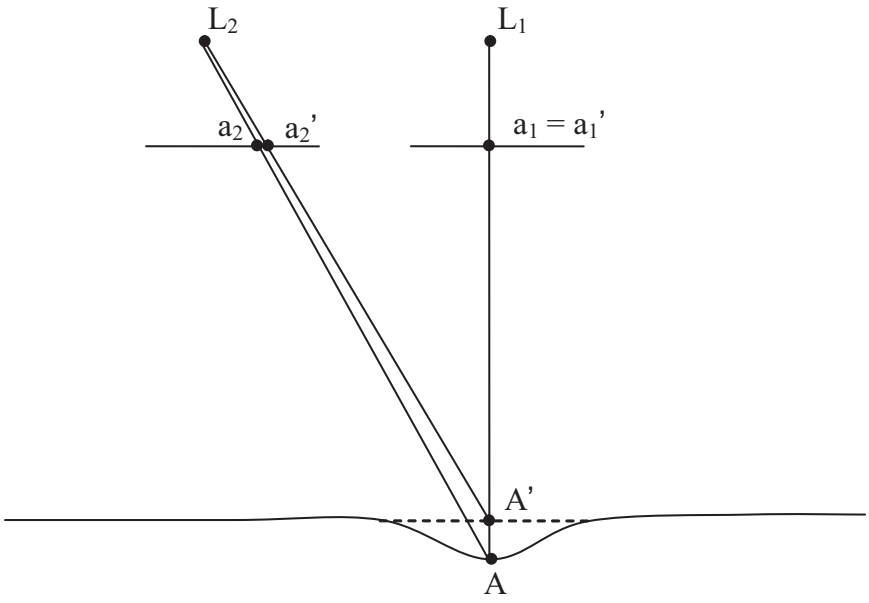

Fig. 8. Example of Disparity due to Depth Damage.

point A in the damaged surface, will map to point a_2' in the tilted image, since the mapping assumes the point is on an undamaged surface. Therefore this will be the starting position for the moving window approach when matching damaged pixels in one image to the same damaged pixels in another image. Once the matching is done the algorithm should find the damaged pixel located at image point a_2. The difference between a_2 and a_2' is called the *pixel disparity* and will be used to determine the depth of the damage at point A.

CHAPTER V

ALGORITHM EXAMPLE

The purpose of this chapter is to outline the specific steps and examples that were used to solve the research problem proposed in Chapter I. The techniques of Chapter IV were used, based upon the methods of Chapters II and III. This example is based upon a test that was performed at NASA's Johnson Space Center (JSC) on June 1, 2005.

A. Background and Setup

The purpose of this test was to validate the system proposed by the STC. The problem, therefore, was simplified somewhat in order to provide a test bed for the system. The nature of the problem outlined in Chapter I Section A outlined the distance of 600 feet from the camera on the ISS to the orbiter. To keep atmospheric effects from disrupting the test, however, it was desirable to have the test indoors. Finding 600 feet of uninterrupted length for which to perform this test is extremely difficult and proved to not exist at JSC. The laser test tunnel, however, provides 200 feet of uninterrupted space. It was decided by the system administrators that this distance would be used for the test and appropriate accommodations made. Therefore a few modifications to the camera had to be made. To provide the same pixel sampling in the test that would be available on orbit, the lens would have to be changed. Using a 600mm lens at 600 feet is virtually identical to using a 200mm lens at 200ft. Therefore a 200mm lens was used in the test. A few modifications to the camera electronics had to be made to accommodate this, however these are not within the scope of this document. Figure 9 shows the length of the tunnel from the perspective of the tiles looking back at the camera.

Fig. 9. Laser Tunnel Used in Test.

In order to test this system, a set of orbiter tiles with simulated damage was used. The individual tiles were situated upon a metal plate. Then the set of tiles was suspended on a movable wooden rack in order to simulate the rotation due to the RPM. This set of tiles is shown in Figure 10. As can be seen from Figure 10, the simulated damage is in a left-right direction. Damage occurring on the bottom of the orbiter during liftoff will usually happen in a nose-to-tail direction, due to the direction of motion and therefore the direction of debris falling off. Therefore this test is simulating the orbiter nose-to-tail direction as on its 'side', in a left-right direction. Because the RPM happens about an axis that is wing-to-wing, this setup makes the rotation of the tiles about a vertical axis, which is easier to perform and easier to measure. Figures 11 - 13 show the measurement scheme by which the angle of tilt was found for the tile orientation to the camera. As can be seen from Figures 11 -

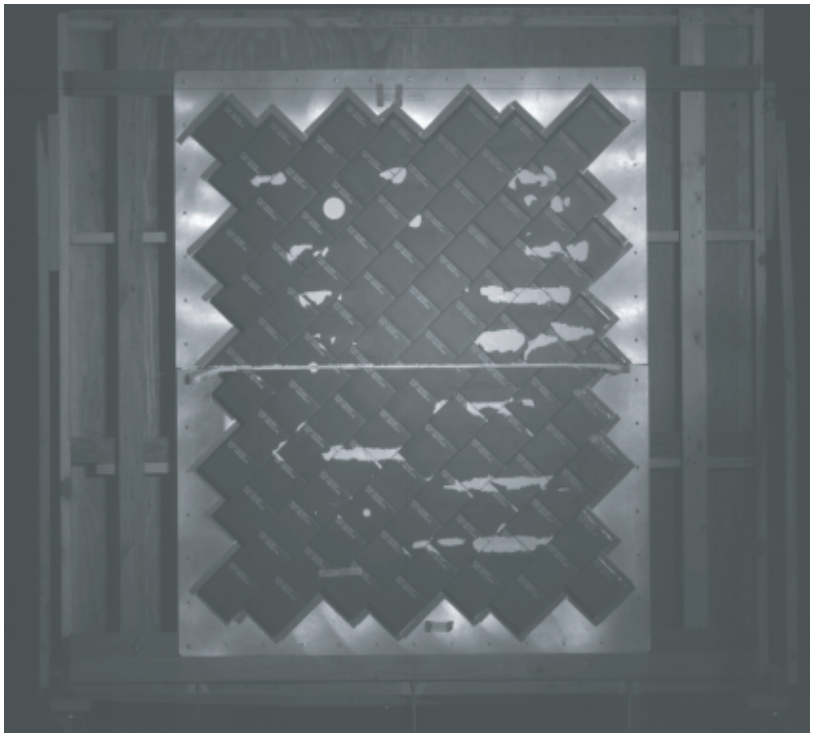

Fig. 10. Orbiter Tile Set Used in Test.

Fig. 11. Measurement Scheme Used in Test, View A.

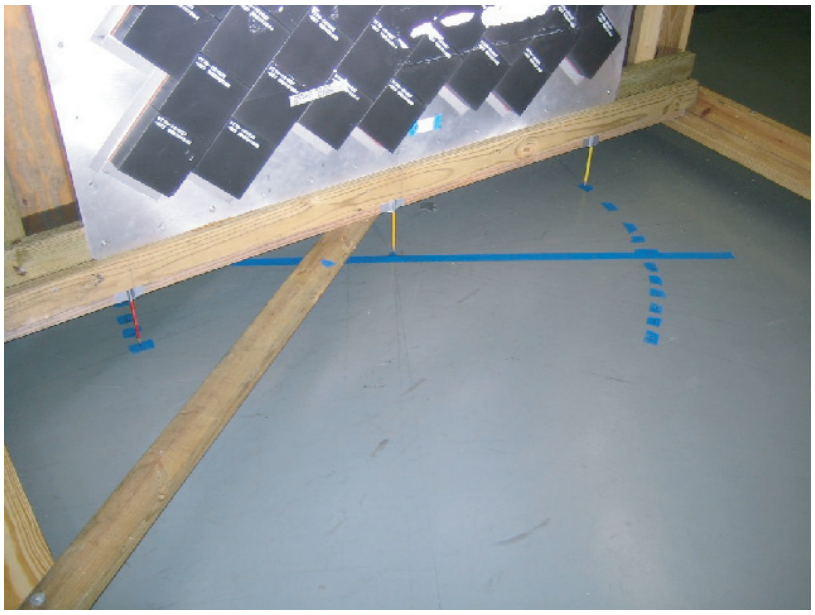

Fig. 12. Measurement Scheme Used in Test, View B.

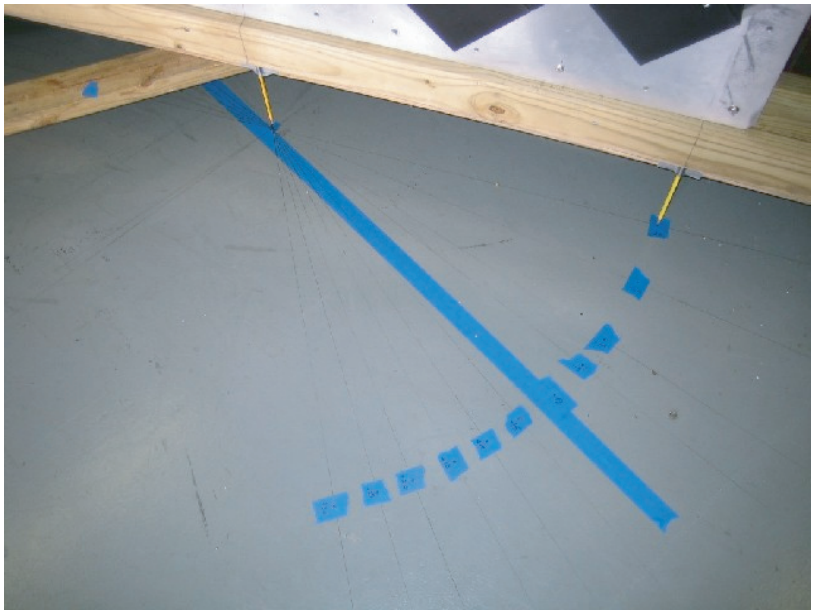

Fig. 13. Measurement Scheme Used in Test, View C.

13, the tiles were rotated through the set of angles given as:

$$[-30°, -20°, -10°, -5°, 0°, 5°, 10°, 20°, 30°]$$

To make sure that the environment used for the test setup would be similar enough to the on-orbit environment, simulating solar lighting had to be used to illuminate the target. Because the test was performed indoors, this required a few large lamps to provide the amount of incident light similar to on-orbit solar lighting. A further problem surfaced in trying to get the lamps close enough to give solar illumination without getting the lamps in the way of the camera. The lighting setup is shown in figure 14.

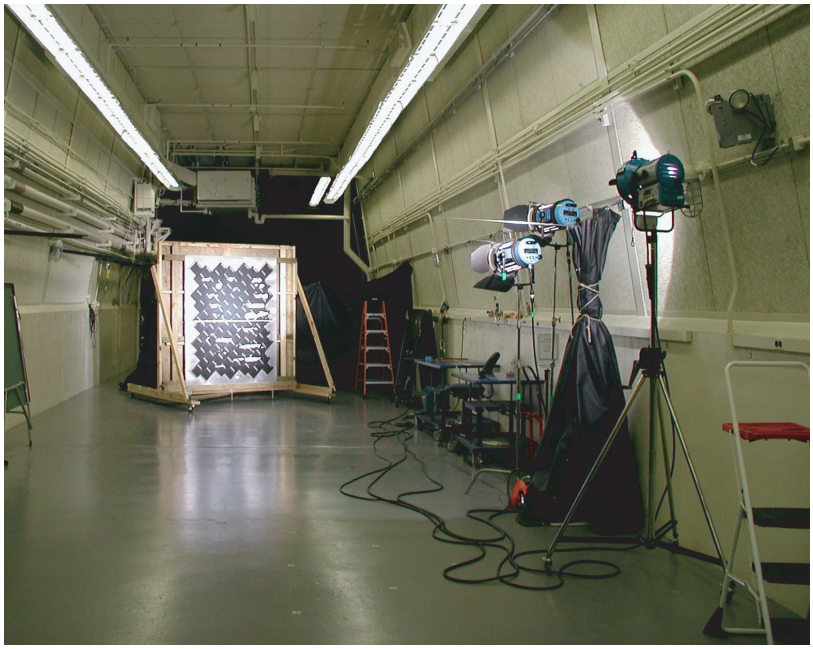

Fig. 14. Lighting Setup Used for Test.

B. Tile Imaging

Another purpose of this test was to determine the camera properties that would be necessary to correctly image the tiles. Therefore a few runs were performed with different camera hardware. The first run was performed without the 1.4x extender. The extender is a piece of optical equipment that zooms in 1.4x in both directions, thereby producing an overall effect of a 2x zoom. Figure 15 shows this run with the full field of view of the camera without the extender. Figure 15 is a vertical image, with the axis of the lens perpendicular to the plane of the tiles. Figure 16 shows a close up of some of the damaged areas for the camera without the 1.4x extender. Figure 16 is simply a cropped version of Figure 15, which allows for more detail to be seen on the page. As was shown in Chapter IV Section C, distinct feature points are necessary on the undamaged plane, which is the black area in Figure 16, in order

Fig. 15. Field of View without 1.4x Extender.

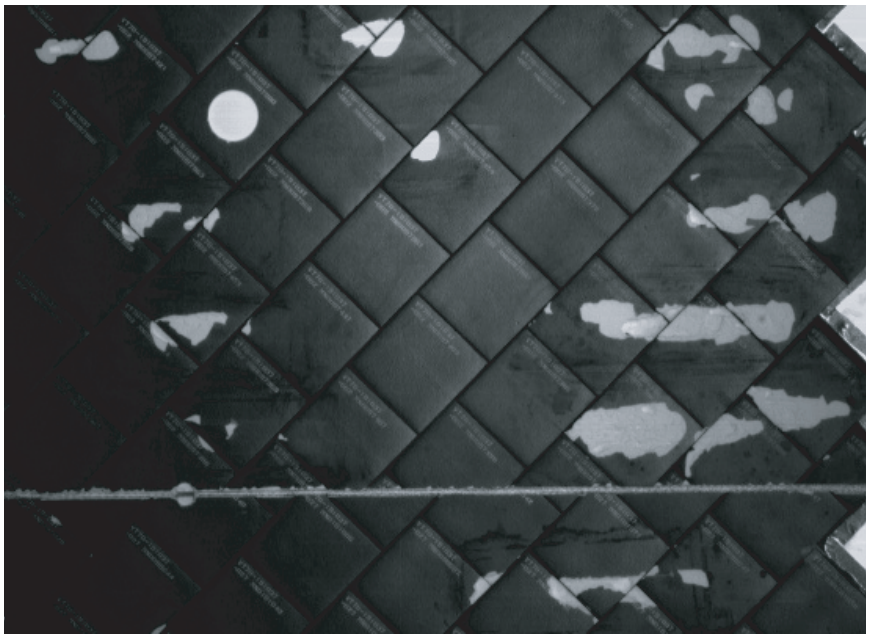

Fig. 16. Tile without 1.4x Extender.

to register the tilted images to the near-vertical images. However, one can notice in Figure 16 that the serial numbers are not distinct and would be very difficult to use. If the serials numbers are blurred, as in this image, then ambiguous matches in the stereo matching used to register would be found, resulting in the images being incorrectly rectified and aligned.

Therefore, a second run was performed with the 1.4x extender. Figure 17 shows the full field of view of the camera with the 1.4x extender. Figure 18 shows a close up of the damaged areas for the camera with the 1.4x extender. Similar to Figure 16, Figure 18 is a cropped version of the full field of view, which is shown for the 1.4x extender in Figure 17. As can be seen from Figure 18, the resolution is higher with the 1.4x extender. The serial numbers on the tiles are distinct and much less likely to have ambiguous matches.

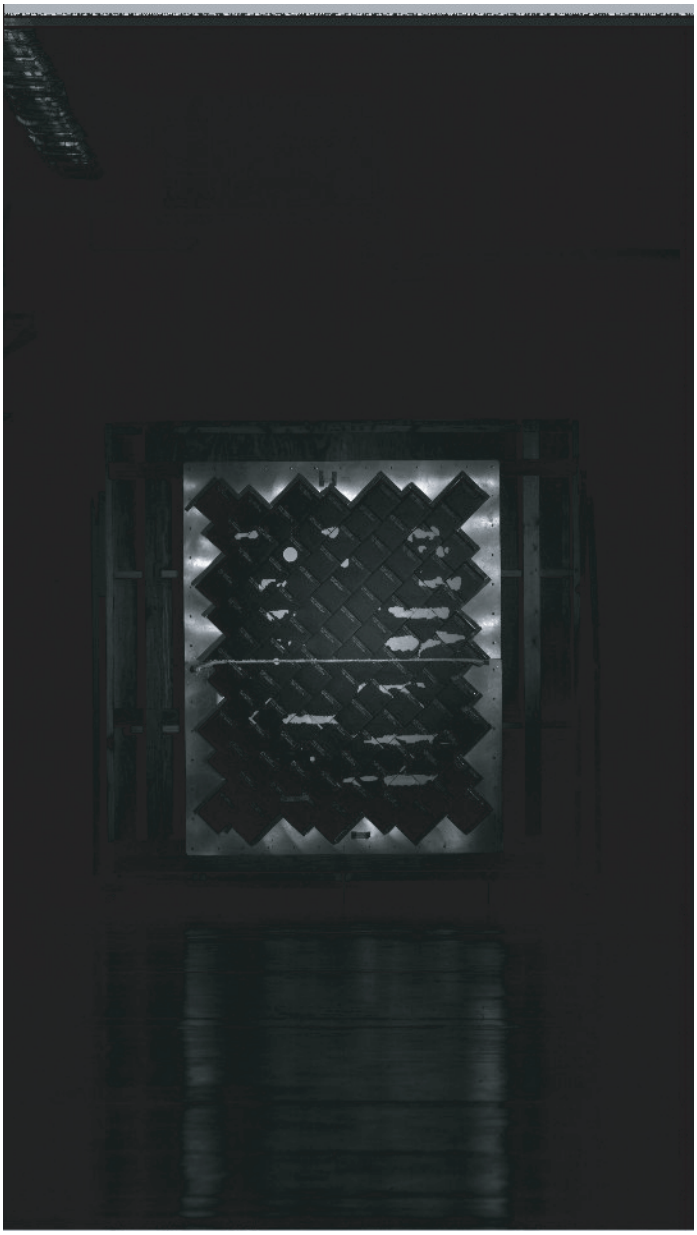

Fig. 17. Field of View with 1.4x Extender.

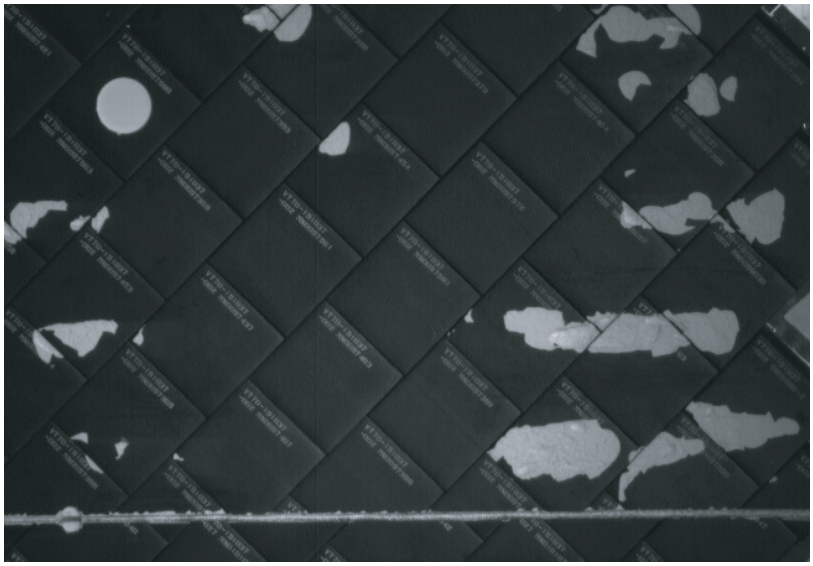

Fig. 18. Tile with 1.4x Extender.

C. Visual Procedure

After the images were taken and stored digitally, they could be operated on by the photogrammetric analysis tools developed through Chapter IV. The first step is to determine the size of the useful area of the image. This was done visually for this test, by inspecting the vertical images (incidence of 0°) and determining the area of usefulness from the images. As can be seen from Figure 15, much of the image that was taken was not useful. The only functional part of the image is the sheet of tiles. Therefore the image was cropped down to useful size, as shown in Figure 19. This portion of the image shows the tile set without the excess data on the sides. Using this pixel boundary set, similar sections of the side tilted images were obtained, thereby keeping the 'cropped' images the same size.

Fig. 19. Useful Segment of Tile without 1.4x Extender.

1. Foreshortening

Before the images can be registered, a few adjustments must be made to ensure that the registration is done correctly. Because the side images are tilted, they do not show exactly the same thing as the vertical image. Consider a pair of images shown in Figure 20. In Figure 20, two images are taken with the same focal length and same size imaging plane. A vertical image, taken from exposure point L_1, has an imaging axis of n_1. The second image is a titled image, taken from exposure point L_2, and has an imaging axis of n_2. The angle between the two axes is given as ϕ. In the vertical image, the edge of the imaging plane exposes object points A and B, with image points a_1 and b_1. In the tilted image, however, object points A and B do not lie at the extent of the image plane, but are foreshortened by the tilt angle, ϕ. The extent of the image plane in the tilted image has image points a' and b', which exposes points A' and B', a larger area than the vertical image. The foreshortening

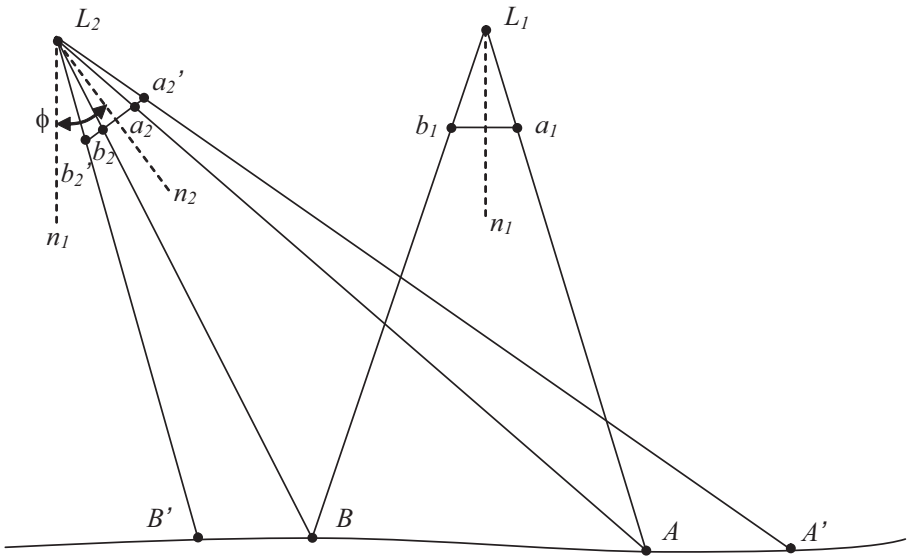

Fig. 20. Foreshortening due to Tilt.

is a $cos\,(\phi)$ effect. This effect is shown in Figure 21, which is a cropped image similar to Figure 19, except it was taken at 20° tilt. As can be seen from Figure 21, which was titled about a 'vertical' axis in the plane of the tiles, there is more area on the sides than in the vertical 'straight-on', vertical incidence image of Figure 19. This can be seen from the extra black past the edge of the silver metal plate.

Therefore the tilted image will show 'more' feature than the vertical image. Due to this, the pixel disparity discussed in Chapter IV Section E will not be accurate. Therefore before the registration is performed, a slight 'stretching' is done so that the tilted image shows the same amount of feature as the vertical image. This will make the registration process easier to perform. Each tilted image is therefore stretched by a factor of $\frac{1}{cos(\phi)}$. This will now make the images larger than the vertical images, therefore the extra pixels added are cut off so that the pixel size of the images stays constant. This process performed on Figure 21 is shown in Figure 22. Comparing

Fig. 21. Example of Foreshortening.

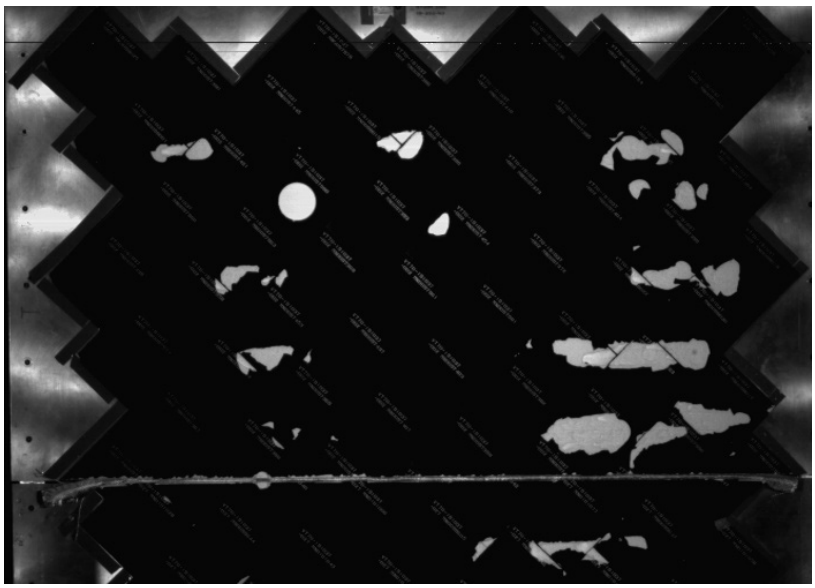

Fig. 22. Example of Stretching to Account for Foreshortening.

Figure 22 to Figure 19 one can see that the edges of both images show roughly the same features, even though Figure 19 is a vertical image, and Figure 22 is a tilted image taken with 20° of tilt.

2. Feature Selection

Once the images cover the same object features, distinct features are chosen visually for the registration. It was explained in Chapter IV Section C that the tilted image would be registered to the vertical image on the undamaged plane. However, it may be necessary to use two tilted images. If this is the case, both images are corrected for foreshortening effects as above. Then the image that is closest to vertical is used as the baseline image, and the image that has a greater tilt angle is registered to it. The photogrammetric analysis is still the same, however the method for determining depth from two tilted images is a little different, as will be shown in Section E1. As can be seen from Figures 19 and 22, the black surface is readily identifiable as the undamaged plane. Also it can be seen that the serial numbers are seen and somewhat distinct. It is these serial numbers, specifically the dashes between the serial numbers, that will be used for this registration.

a. Loose Registration

First, three points are selected in both the vertical image and the tilted image. The same three points are selected in both, to perform a loose 'mapping' from the vertical image to the tilted image. A sample of three points that were selected in each image is shown in Figures 23 and 24. Though it may be difficult to see, the features selected are the dashes in the serial numbers. It was noted above that the serials numbers were difficult to see in the images without the 1.4x extender, and this fact can be seen in Figures 23 and 24.

Fig. 23. Example of Feature Selection, Vertical Image.

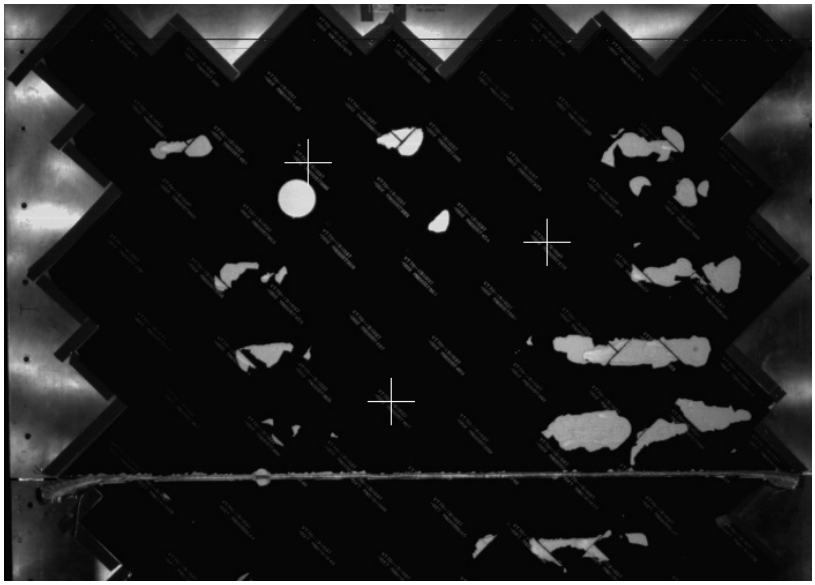

Fig. 24. Example of Feature Selection, Tilted Image.

Once the features are selected, a loose registration is performed to align these areas to each other. The purpose of this loose registration is to provide a loose alignment of the images. It is desired that the pixel locations, or image points, of object features in the vertical image be very close to the pixel locations of the same object features in the tilted image. The row and column pixel indices of these points in the tilted image are mapped to the row and column pixel indices of these points in the vertical image through a linear regression similar to the form of Equations 4.1 and 4.2. Once this regression is performed, the transformed indices of the tilted image are found through the regression variables by Equations 5.1 and 5.2

$$ii = c_x + r_{x,i} * i + r_{x,j} * j \tag{5.1}$$

$$jj = c_y + r_{y,i} * i + r_{y,j} * j \tag{5.2}$$

where, i and j are the original, whole number column and row pixel indices, respectively, in the tilted image; c_x, $r_{x,i}$, and $r_{x,j}$ are the regression coefficients of the column (or i) indices; c_y, $r_{y,i}$, and $r_{y,j}$ are the regression coefficients of the row (or j) indices; ii are the new, fractional column indices of the tilted image; and jj are the new, fractional row indices of the tilted image. As can be seen from Equations 5.1 and 5.2, a multi-variable regression was performed, mapping the original row and column indices from the vertical image first to the column indices of the tilted image, and then to the row indices of the tilted image. To determine the pixel values at the new fractional indices, the tilted image was resampled using the bilinear method shown in Chapter III Section B1.

Now that the tilted image has been resampled, the images are loosely aligned. This means that if the three feature points selected above lie at pixel locations (i_1, j_1), (i_2, j_2), and (i_3, j_3) in the vertical image, they should lie at the same pixel locations

in the tilted image. However, this does not ensure that every point in the image is aligned correctly. Therefore a fine registration was performed on the images.

b. Fine Registration

The fine registration is similar to the loose registration with a few modifications. First, the fine registration will employ a larger set of points to ensure proper registration. Second, where the loose registration relied solely on the visual inspection and 'clicking' of the user, the fine registration will employ a more thorough approach.

Just like in the loose registration, the feature points used in the fine registration are the dashes in the serial numbers on the tiles. Figure 25 shows the feature points used for a typical fine registration.

Fig. 25. Example of Fine Registration Feature Selection, Vertical Image.

Before the loose registration, it could not be ensured that features in one image were located at the same pixel locations in the other image. Therefore any automated

matching could not be performed with any assurance of no false positives. Therefore the loose registration relied on the user clicking on the same feature point in both images. However, the loose registration provided a parametric mapping of a few feature points so that one can be relatively assured that image points in one image are located close to the same pixel location in another image. Therefore, the feature points are selected only in the vertical image. Then they are found in the tilted image through an automated matching using the methods of Chapter IV Section D2b.

A few differences from these methods, however, are necessary. Because the images are not yet relatively aligned, a two-dimensional search is necessary in the tilted (or slave) image. An area of a 20 by 20 pixel window centered about the feature point 'clicked' by the user in the vertical (master) image was used as the area to match to the slave (tilted) image. This box was slid through an area of 20 pixels in both dimensions in the slave image, using a least squares approach to match the slave image to the master. Also, since the point where the user clicks may not be exactly on the serial number, a weight matrix is not used. This matches the entire area of the box equally in both images.

Up to this point, images without the 1.4x extender have been used in the examples. However, it is at this point that the necessity of the 1.4x extender becomes clear. Because the dashes and serial numbers are not distinct in the images without the 1.4x extender, a few problems arose. For some of the images, the stereo matching was unable to perform a fine registration correctly. Because the features are not distinct, ambiguous matches were found and the alignment was not correct. However, even for the images where the fine registration was able to align correctly, the pixel sampling was such that the resolution requirement was not met. Therefore from this point on images with the 1.4x extender will be used. Similar to Figure 25, the feature points used for the fine registration for the images with the 1.4x extender are shown

in Figure 26.

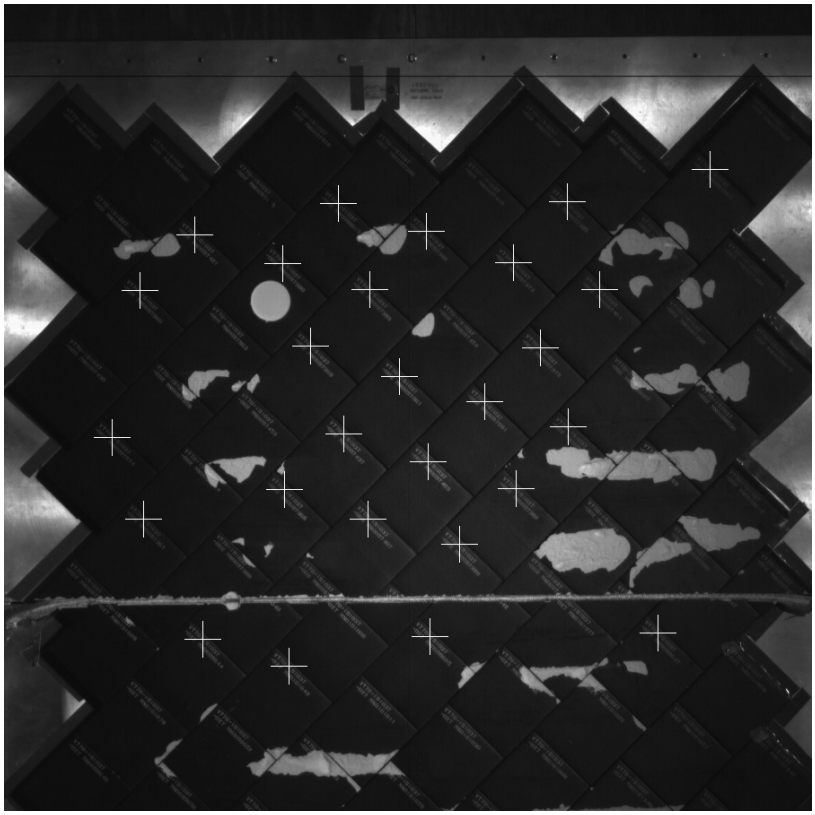

Fig. 26. Example of Fine Registration Feature Selection, Vertical Image with 1.4x Extender.

In order to aid the stereo matching algorithm, the visually striking features were enhanced using an edge detection algorithm shown in Chapter III Section C. This enhancement is shown in Figure 27 which includes the 1.4x extender. The stereo matching search was performed on these edge enhanced images for a less likely possibility of ambiguous matching. It is to be noted that even with this edge enhancement, the fine registration failed on the images without the 1.4x extender. It can be seen from Figure 27 that the serial numbers are even more distinct in the edge enhanced images. Once the match is complete for each of the feature points, the image points

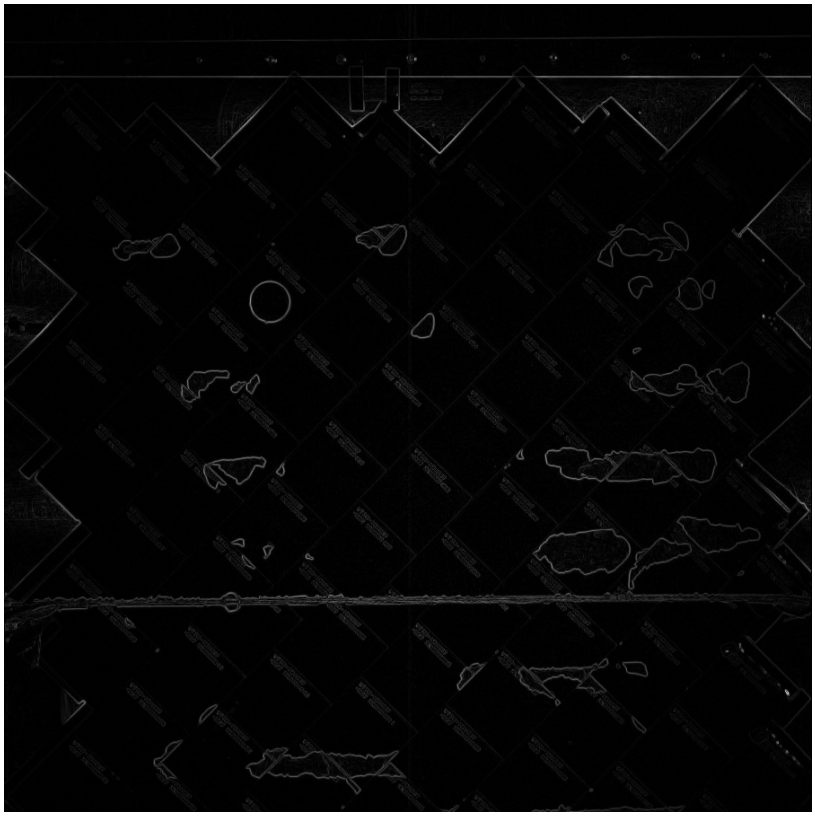

Fig. 27. Edge Enhancement for Fine Registration, Vertical Image.

in the tilted image that correspond to the same feature in the vertical image will be known. These image points, or column and row indices, in the tilted image will then be mapped through a linear regression similar to the form of Equations 4.3 and 4.4. Once the regression is complete, the new row and column indices of the tilted image will be found through the regression variables by Equations 5.3 and 5.4

$$ii = c_x + r_{x,i} * i + r_{x,j} * j + r_{x,i^2} * i^2 + r_{x,j^2} * j^2 \tag{5.3}$$

$$jj = c_y + r_{y,i} * i + r_{y,j} * j + r_{y,i^2} * i^2 + r_{y,j^2} * j^2 \tag{5.4}$$

where i and j are the original, whole number column and row pixel indices, respectively, in the tilted image; c_x, $r_{x,i}$, $r_{x,j}$, r_{x,i^2}, and r_{x,j^2} are the regression coefficients of the column (or i) indices; c_y, $r_{y,i}$, $r_{y,j}$, r_{y,i^2}, and r_{y,j^2} are the regression coefficients of the row (or j) indices; ii are the new, fractional column indices of the tilted image; and jj are the new, fractional row indices of the tilted image. Once again, a multivariable regression was performed, mapping the original row and column indices from the vertical image first to the column indices of the tilted image, and then to the row indices of the tilted image. The image was then resampled using the bilinear method shown in Chapter III Section B1.

Prior to fine registration, the images are only loosely aligned. A superposition of the vertical and tilted images is shown in Figure 28, with the vertical image in normal black and white, and the tilted image enhanced in red. For a better description, Figure 29 shows the same superposition with the vertical edge enhanced image in white and the tilted edge enhanced image in red.

One will notice in Figures 28 and 29 that on the damage and serial numbers near the edges of the image, the images are not aligned properly. This is due to the fact that the feature points used for the loose registration were near the center of the

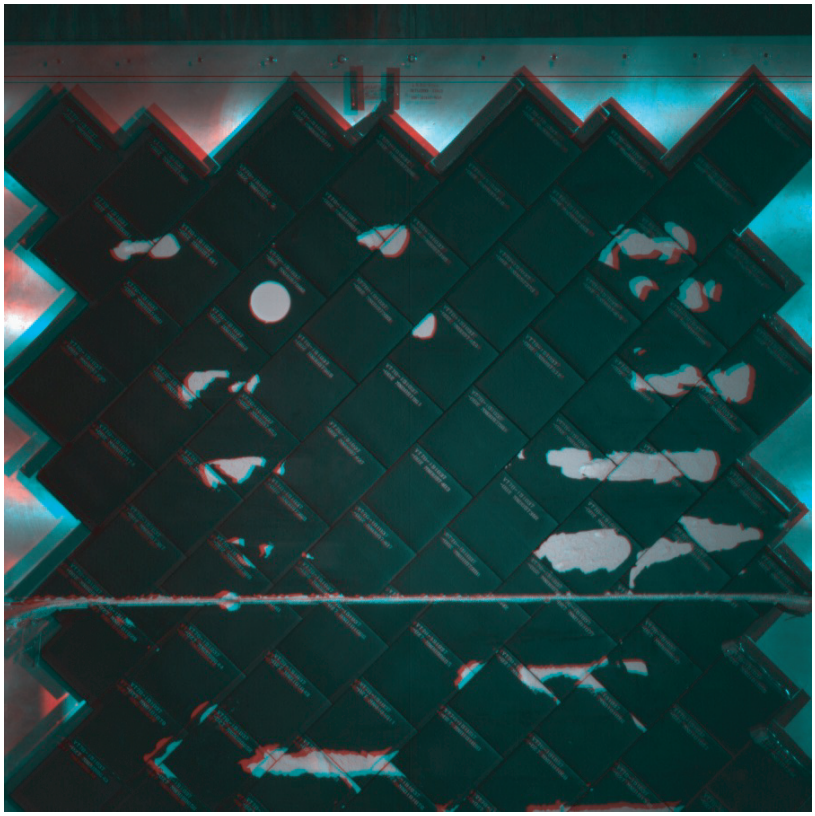

Fig. 28. Pre-registration Superposition.

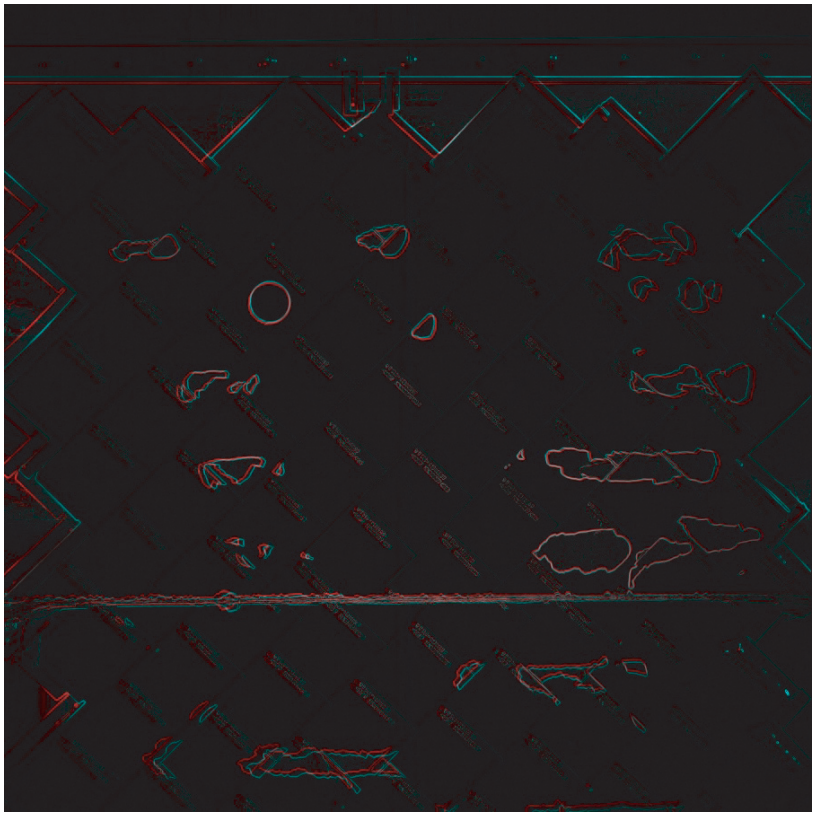

Fig. 29. Pre-registration Superposition, Edge Enhanced.

image. In choosing the feature points to use for the registration, it is best to choose points located around the damage that is being sought.

Once the fine registration is complete, the tilted images are relatively aligned to the vertical image. A superposition of the registered tilted edge enhanced image on the edge enhanced vertical image, similar to Figure 29, is shown in Figure 30. It

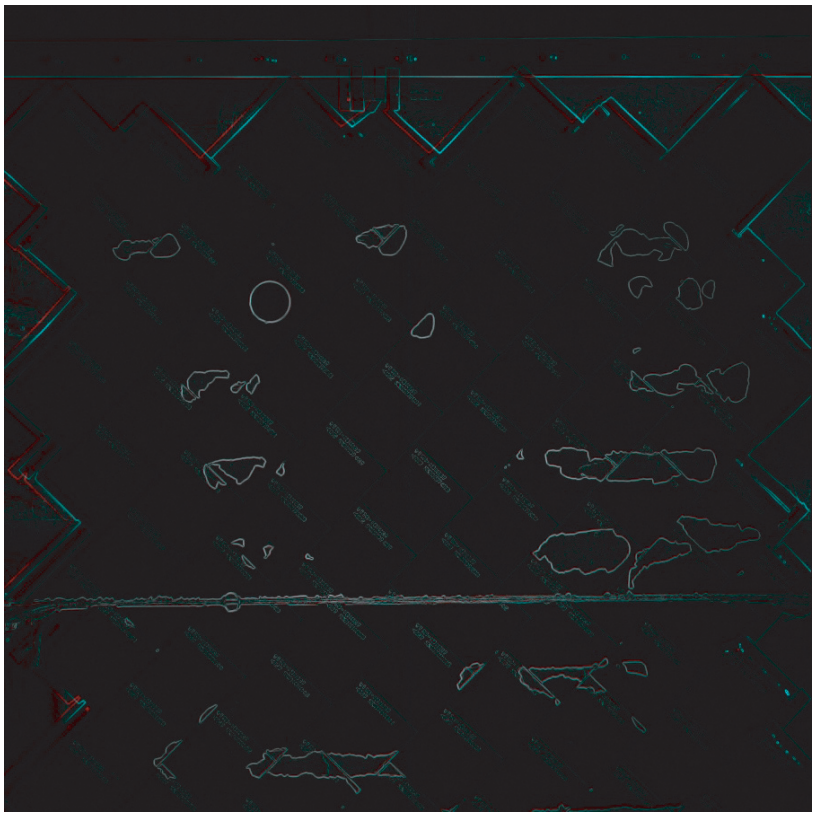

Fig. 30. Fine Registration Superposition, Edge Enhanced.

can be seen from Figure 30 that the serial numbers and edges of the damage are all aligned. It may be hard to see the red overlay on these feature points, but that is exactly what is desired. The red and blue overlay show up as a teal in Figure 30 since they lie in the same spot on the image. The overlay can be seen, however, near the top of the image where the edges of the tiles could not be registered. This is due to

the fact that the tiles are very thick, and show another edge when tilted. This will not be a problem in the final version, because all the tiles on the orbiter will butt up against one another, leaving no tile edges to see.

D. Disparity Search

Now that the tilted images are relatively oriented to the vertical images, it can be ensured that the epipolar lines are parallel to each other, per Chapter IV Section C1. It is also certain that feature points that lie on the undamaged plane located in the vertical image lie at the same pixel location in the registered tilted image. The only features that should not lie in the same pixel location in both images are pixels that lie in the damaged areas. Points that lie in the damaged areas in the vertical image, however, are still ensured to lie in the same row in the tilted image, because the rows are the epipolar lines after the images are relatively oriented. Therefore comparisons between the two images can be made on a pixel by pixel basis.

1. Lighting Issues

The tiles are very reflective of light. This can pose a problem when trying to match items under different lighting conditions. Specifically, the tiles are highly susceptible to specularity when under direct incident light. This can be seen from Figure 31, taken at 10° tilt. As can be seen from Figure 31, the specular reflection washes out many features, especially the serial numbers on the tiles. Therefore the highly specular images are unable to be registered and cannot be used. Different lighting incidences, however, also provide different illumination levels, and therefore different gray levels in the images. Because damage to the tiles will be holes with a non-uniform surface, the texture of the damage will cast small shadows on itself. However, it is difficult

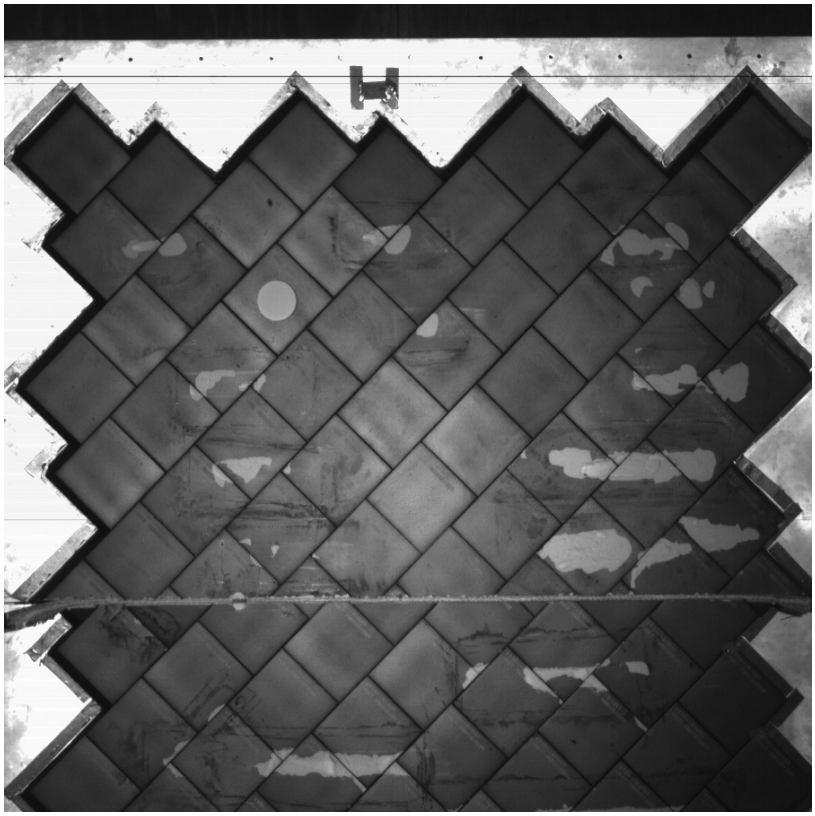

Fig. 31. Specular Reflection from Lighting Source, 10° Tilt.

to match to the texture or the shadows if they have different gray levels. Therefore, because the images can now be compared on a pixel by pixel basis, ensuring that the same features are being looked at in both images, the gray levels can be compared.

The registered tilted images are first fit to the vertical-incidence image through a simple linear regression of the form given in Equation 5.5.

$$y = A * x + b \tag{5.5}$$

The difference between this regression and the previous fits used for registration is that the previous fits were for row and column indices to align the images. This regression is for the pixel values themselves. Therefore the independent variable will be the pixel values in the registered tilted image, and the dependent variable will be the corresponding pixel values in the vertical-incidence image. This will 'cast' the tilted image gray values into the vertical image gray values. This regression is only performed in this test, however, on the relevant pixel locations. Therefore, pixels that lie off the tiles are not considered. An example of the pixel values used is shown in Figure 32. It can be seen from Figure 32 that the pixels used in the regression constitute a good subset of the tile. Most of the damage is included, specifically all the relevant damage that is desired to be studied. Also, pixels that lie off the tiles are not considered.

Once this fit it accomplished, the tilted image is transformed through Equation 5.6

$$im_{1,new} = A * im_{1,old} + b \tag{5.6}$$

where im_1 is the registered tilted image. Through this fit, the relative difference in gray levels between the pixel values in the registered tilted image has not changed; it has simply been scaled to match the gray levels of the vertical image so that a

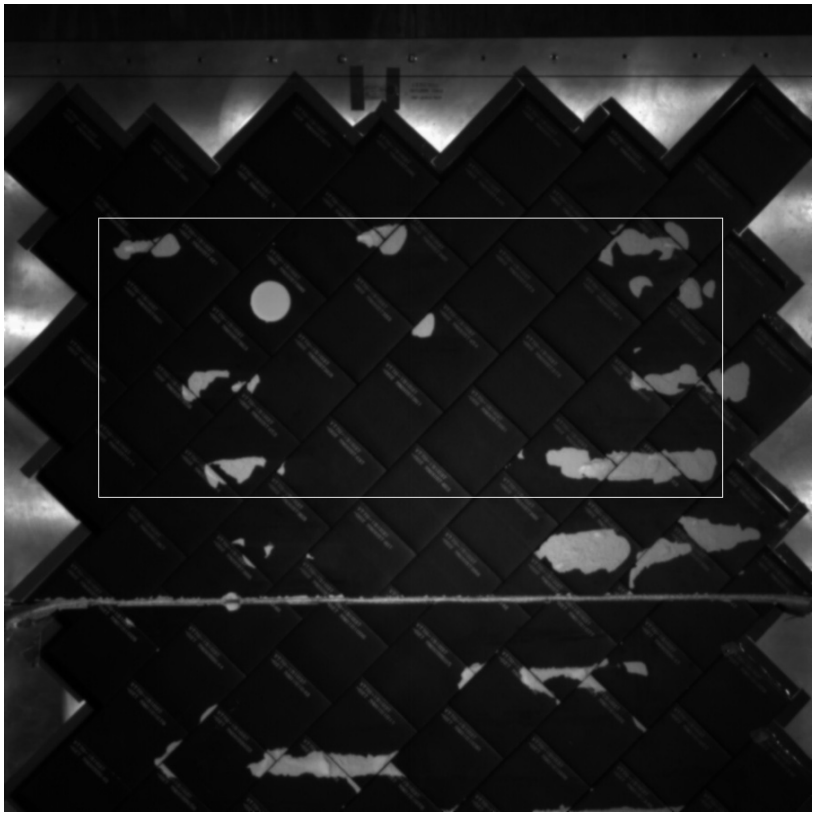

Fig. 32. Pixel Limits Used for Regression.

comparison can be made between the two with better accuracy.

2. Damage Identification

Next, it is not necessary to match every pixel in the vertical image to its corresponding pixel in the registered tilted image. In fact, this is what was accomplished for the undamaged plane in the registration process. Therefore, the disparity search should only be performed on pixels that lie in the damaged portions of the tile. Nothing is lost if the search is performed on undamaged tile, however it is computationally inefficient. It can be noticed in the previous images that the undamaged surface of the tile is black and the damaged portions are white. Therefore the search only need to be performed on the 'white' areas of the tile. It should not be simply the white pixels that are searched, however, because many of these areas are serial numbers, or the edges of the damage. A true/false map was therefore created using a 'smoothed' vertical image. This smoothing is a low pass filter, or simply an averaging of the pixels in a box centered about each pixel in the image. For this process, a 9 x 9 pixel box was used. The low-pass filter results in a blurring effect, obscuring sharp features like the serial numbers and the edges of the damage. An example of this smoothing is shown in Figure 33. Once the image has been smoothed, a true/false map is created where the pixels above a certain value are given a value of 1, and the pixels below a certain value are given a value of 0. This map gives the search region for the disparity search, where pixel locations that were given a value of 1, or $true$, are searched for disparity. Pixel locations that were given a value 0, or $false$, are assumed to be undamaged tile and not searched. As can be seen from Figure 33, the boundaries of the damaged areas are blurred and have increased in size. This smoothing, along with the corresponding truth map, ensures that the entire damaged area is searched. An example of the truth map is shown in Figure 34. It can be seen

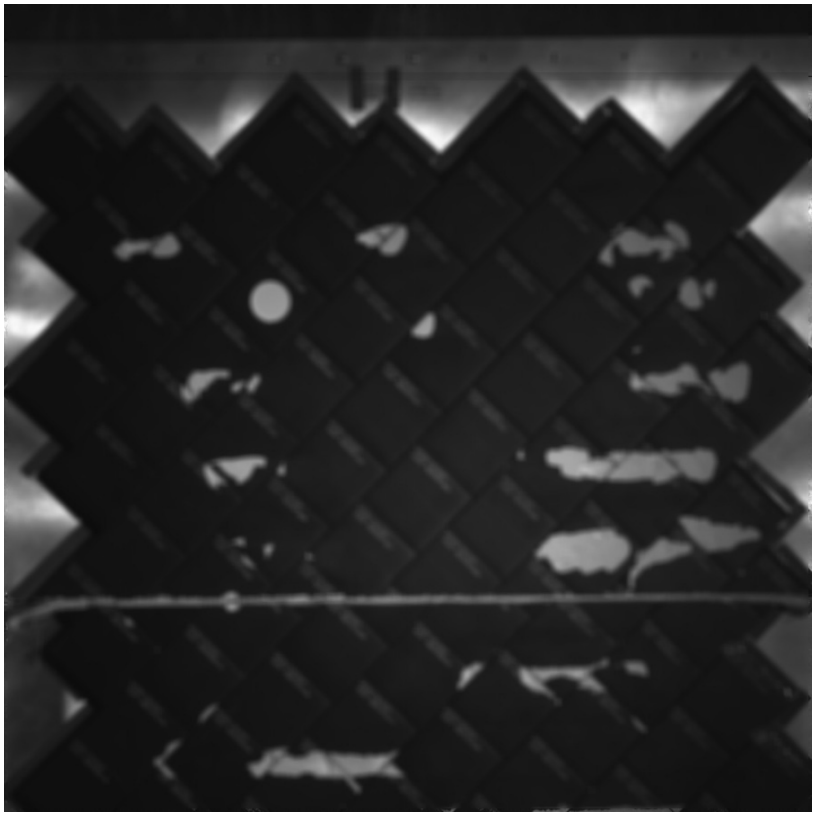

Fig. 33. Example of Low Pass Filter, 9x9 Smoothing.

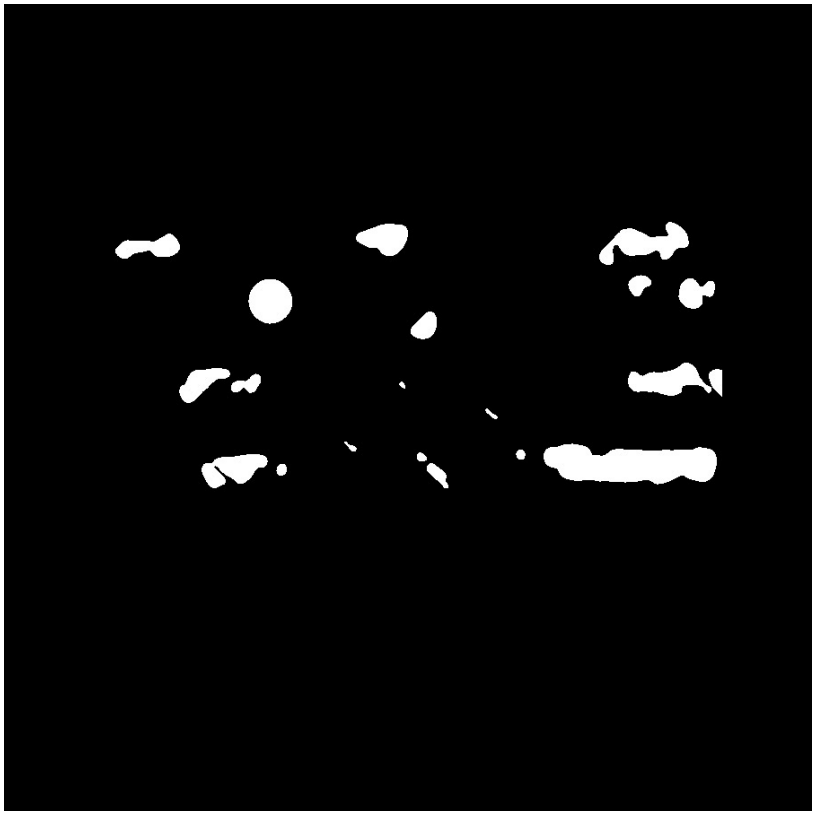

Fig. 34. Example of True/False Map for Disparity Search.

from Figure 34 that all the pixels that lie outside the box outlined in Figure 32 are given a value of false, because they are not to be considered anyways. The damaged areas inside this box are given a value of true, along with some of the serial numbers. Most of the serial numbers were given false values, due to the low-pass filter losing the detail of the white in the surrounding black tile. The serial number areas that are left are acceptable, because if the cutoff value for the truth map is raised so high as to give the serial numbers a false value, then it is likely that pixels in the damaged areas may have similar gray values and also be given a false value, thereby precluding them from the disparity search. Therefore some of the serial numbers will be searched for disparity and their matches will be disregarded by the user through inspection. It is also noticed that the true values in the map for the damaged areas are larger than the damaged areas themselves, as was shown earlier. This is desirable and is a result of the low-pass filter. It is advantageous to search for disparity in a region larger than the damage, to ensure that the entire damaged region is searched.

3. Window Shifting for Area Matching

All that is left is to do the actual disparity search on the damaged areas. For each of the pixels that lie in the truth area of Figure 34 a 13 column x 7 row pixel window is made around it to area match. First, a weight matrix of the same size is created. The weight matrix used is an exponential decay matrix and will weight the edges of the area box the same. First a 13 x 7 matrix, x_r, is created with every row equal and given as:

$$\begin{bmatrix} 2.00 & 1.67 & 1.33 & 1.00 & 0.67 & 0.33 & 0.00 & 0.33 & 0.67 & 1.00 & 1.33 & 1.67 & 2.00 \end{bmatrix}$$

Next, a 13 x 7 matrix, y_c, is made with every column equal and given as:

$$\begin{bmatrix} 2.000 \\ 1.333 \\ 0.667 \\ 0.000 \\ 0.667 \\ 1.333 \\ 2.000 \end{bmatrix}$$

Next, each entry in the 13 x 7 weight matrix is given by Equation 5.7, where both the exponentiation and the square are on an element-by-element basis in the matrices x_r and y_c.

$$wt\,(i,j) = e^{-x_r(i,j)^2 - y_c(i,j)^2} \tag{5.7}$$

Finally the weight matrix is normalized by the sum total to ensure no scaling effects.

As was stated above, each pixel that lies in the truth area of Figure 34 in the near-vertical image will be matched to a pixel in the registered tilted image that corresponds to the same features. A 13 x 7 pixel window, or subset of the image, is taken surrounding this pixel. Next, a 13 x 7 window, or image subset, is made around the corresponding pixel location in the registered tilted image. Figure 35 is an example of this procedure in the master image. The numbers are the pixel values at each location in the master image. The pixel to be identified is the bold **21**. The 13 x 7 window is shown as the line outlining the pixel values. Figure 36 shows the same procedure for the slave image. The 13 x 7 window is centered around the same pixel location in the slave image, which in this case is the bold **23**. However, it is still likely that some of the pixels in one image are under different lighting conditions than the same pixel area in the other image. Therefore each value in the window

39	38	36	33	30	27	25	24	23	22	21	19	17	16	14	13	13
37	35	32	29	27	26	24	23	22	21	19	17	16	14	13	13	13
35	32	29	27	25	25	24	22	21	19	17	16	14	13	13	12	12
33	29	27	25	25	24	22	**21**	19	17	16	14	13	13	12	12	12
30	27	25	24	24	23	21	19	17	16	15	14	13	13	12	12	12
28	26	24	24	23	21	19	17	16	15	14	13	13	13	12	12	12
26	25	24	23	21	20	17	16	15	14	13	13	13	13	12	12	12

Fig. 35. Example of 13 x 7 Window for Master Image.

27	27	27	27	26	26	27	25	25	25	23	21	18	15	14	14	13
27	27	26	25	26	26	26	26	24	23	21	19	16	13	13	13	14
27	27	26	26	25	26	25	24	23	21	18	16	14	13	14	13	14
27	27	27	27	26	26	24	**23**	21	18	15	14	14	14	13	13	14
27	27	27	27	26	25	23	21	18	15	14	14	13	14	14	13	14
27	27	26	25	24	23	20	17	15	14	14	13	13	14	14	14	15
27	26	25	24	23	21	18	14	14	13	13	14	13	13	14	15	14

Fig. 36. Example of 13 x 7 Window for Slave Image.

in the master, or vertical, image is divided by the pixel value that the window is centered about. This can be seen in Figure 37, where only the window of Figure 35 is reproduced. Each value in the master window has been divided by the value in the center, or **21**. Similarly, each pixel in the slave window is divided by the pixel value that the box is centered about. Figure 38 shows this for the window in Figure 36. The pixel value in the center of the window in both images should obviously be

1.667	1.524	1.381	1.286	1.238	1.143	1.095	1.048	1	0.905	0.81	0.762	0.667
1.524	1.381	1.286	1.19	1.19	1.143	1.048	1	0.905	0.81	0.762	0.667	0.619
1.381	1.286	1.19	1.19	1.143	1.048	**1**	0.905	0.81	0.762	0.667	0.619	0.619
1.286	1.19	1.143	1.143	1.095	1	0.905	0.81	0.762	0.714	0.667	0.619	0.619
1.238	1.143	1.143	1.095	1	0.905	0.81	0.762	0.714	0.667	0.619	0.619	0.619

Fig. 37. Example of Window Scaling for Master Image.

1.174	1.13	1.087	1.13	1.13	1.13	1.13	1.043	1	0.913	0.826	0.696	0.565
1.174	1.13	1.13	1.087	1.13	1.087	1.043	1	0.913	0.783	0.696	0.609	0.565
1.174	1.174	1.174	1.13	1.13	1.043	**1**	0.913	0.783	0.652	0.609	0.609	0.609
1.174	1.174	1.174	1.13	1.087	1	0.913	0.783	0.652	0.609	0.609	0.565	0.609
1.174	1.13	1.087	1.043	1	0.87	0.739	0.652	0.609	0.609	0.565	0.565	0.609

Fig. 38. Example of Window Scaling for Slave Image.

1, and therefore will not provide any comparison between the two images. Therefore the pixel location at the center of the window in the slave image is replaced by the ratio of the center pixel value from the slave window to the center pixel value in the master window. The center pixel, being the most heavily weighted from the weight matrix used, gives a reference to make sure that the pixel you are searching for is very similar to the same one in the master image. For the above example, this would mean that the center value for the scaled slave window is replaced by 23/21. This can be seen in Figure 39, where the **1** from Figure 38 has been replaced by the ratio

above.

1.174	1.13	1.087	1.13	1.13	1.13	1.13	1.043	1	0.913	0.826	0.696	0.565
1.174	1.13	1.13	1.087	1.13	1.087	1.043	1	0.913	0.783	0.696	0.609	0.565
1.174	1.174	1.174	1.13	1.13	1.043	**1.095**	0.913	0.783	0.652	0.609	0.609	0.609
1.174	1.174	1.174	1.13	1.087	1	0.913	0.783	0.652	0.609	0.609	0.565	0.609
1.174	1.13	1.087	1.043	1	0.87	0.739	0.652	0.609	0.609	0.565	0.565	0.609

Fig. 39. Example of Window Scaling for Center Value in Slave Image.

The window in the slave image will be slid along the row, or epipolar line since the images are registered together, and the new window in the slave image will also be compared to the stationary window in the master image. This is shown in Figure 40, where the window of Figure 36 has been slid one pixel to the right and is now centered upon the bold **21**. This window will also be scaled similar to Figure 39, and is shown in Figure 41. It can be noticed that the center value in Figure 41 is still

27	27	27	27	26	26	27	25	25	25	23	21	18	15	14	14	13
27	27	26	25	26	26	26	26	24	23	21	19	16	13	13	13	14
27	27	26	26	25	26	25	24	23	21	18	16	14	13	14	13	14
27	27	27	27	26	26	24	23	**21**	18	15	14	14	14	13	13	14
27	27	27	27	26	25	23	21	18	15	14	14	13	14	14	13	14
27	27	26	25	24	23	20	17	15	14	14	13	13	14	14	14	15
27	26	25	24	23	21	18	14	14	13	13	14	13	13	14	15	14

Fig. 40. Example of Window Shift for Slave Image.

1.238	1.19	1.238	1.238	1.238	1.238	1.143	1.095	1	0.905	0.762	0.619	0.619
1.238	1.238	1.19	1.238	1.19	1.143	1.095	1	0.857	0.762	0.667	0.619	0.667
1.286	1.286	1.238	1.238	1.143	1.095	**1**	0.857	0.714	0.667	0.667	0.667	0.619
1.286	1.286	1.238	1.19	1.095	1	0.857	0.714	0.667	0.667	0.619	0.667	0.667
1.238	1.19	1.143	1.095	0.952	0.81	0.714	0.667	0.667	0.619	0.619	0.667	0.667

Fig. 41. Example of Window Shift and Scaling for Slave Image.

1, however this is due simply to the fact that the center values in the master window and the new slave window are both **21**.

It is desired to perform a least squares matching as the slave window is slid along the epipolar line. This matching will seek to minimize Equation 5.8, where win_1 is the stationary window in the master image, and win_2 is the k^{th} window in the slave image [14].

$$err\,(k) = \sum_i \sum_j wt(i,j) * (win_1(i,j) - win_2(i,j,k))^2 \tag{5.8}$$

A new value for *err* will be found for each new window location in the slave image. Because the *err* value in Equation 5.8 is the sum square difference between the two windows, the least value will be the window location in the slave image that is the best fit.

The window in the slave image will be slid in the direction of the anticipated disparity shift. Depending on the rotation, positive or negative, the shift will either be left or rightward due to depth damage. Typically, the shift from the first centered pixel in the slave image in the direction of the rotation is from -2 to 8 pixels, in whole pixel increments. Starting at -2 ensures that if the damage caused the tile to actually extrude above the damage plane, it will still be found. The 8 pixels in the direction of expected disparity shift is due to the expected damage depths of about 0.25 inches to about 2 inches.

Therefore there will be 11 values for *err* in Equation 5.8, one for each of the window locations in the slave image. The smallest *err* value will correspond to the best whole pixel slave window shift match to the master window. This location should be very close to the location a_2 in Figure 8. In order to find the best match, however, sub-pixel shift accuracy is preferred. This can be shown in Figure 42, where the locations on a curve for each of the pixel window shift values are shown. It can be

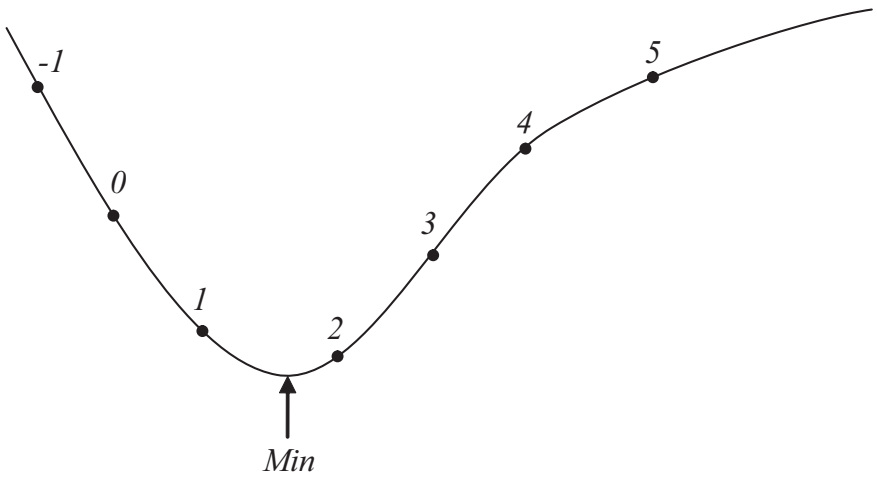

Fig. 42. Example of Minimum Pixel Window Shift.

seen that the minimum whole pixel shift value is 2, however this is not the minimum of the curve. To find the minimum, therefore, a 2^{nd} order polynomial fit for the data is found. The minimum whole number value and the value on either side are used. For the example in Figure 42, this would be the points located at 1, 2, and 3. For each of these points a 2nd order polynomial can be written as Equation 5.9.

$$f(x_i) = a * x_i^2 + b * x_i + c \tag{5.9}$$

These three equations are put in matrix form as shown in Equation 5.10 and in compact form in Equation 5.11.

$$\begin{bmatrix} f(x_1) \\ f(x_2) \\ f(x_3) \end{bmatrix} = \begin{bmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ x_3^2 & x_3 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} \tag{5.10}$$

$$\underline{f} = H\underline{x} \tag{5.11}$$

Then the values for the polynomial constants, $\underline{x}$, can be found quite easily. This will fit a curve to these three values, and the minimum is found at the location shown in Figure 42 as $\frac{-b}{2a}$ [15].

Another feature of fitting a polynomial to the data is that the curvature, a, gives a good measure as to the certainty of the minimum. If the curvature is high, then it is more likely that the match is good. If the curvature is low, however, the data lie on a flatter line, and it is likely that the match is not very good.

E. Disparity Results

The regions of interest found in the truth map of Figure 34 are shown in Figure 43. These regions were noted by NASA as areas for the proof-of-concept test. The pixel disparity for these regions are shown in Figure 44, where damage is shown positive for clarity. The image tilt angles used to determine these disparities were 5° and 20°.

It can be seen in Figure 44 that the areas shown in Figure 43 show considerable damage. It is to be noted that the large spike at the top left of Figure 44 is the circular hole that was cut out in Figure 43. Since this hole was machined out, there was very little texture for the area search to find and is not a good fit. It can also be noted that damage such as this will not be found on orbit.

Pixel disparity for each of the regions shown in Figure 43 are shown in detail in Figures 45 - 52.

1. Damage Depth Determination

Once the sub-pixel disparity shift has been found, all that is left is to convert the pixel disparity shift to depth. Figure 53 is similar to Figure 8. In Figure 53, a vertical image taken at exposure station L_1 and a tilted image taken at exposure station L_2

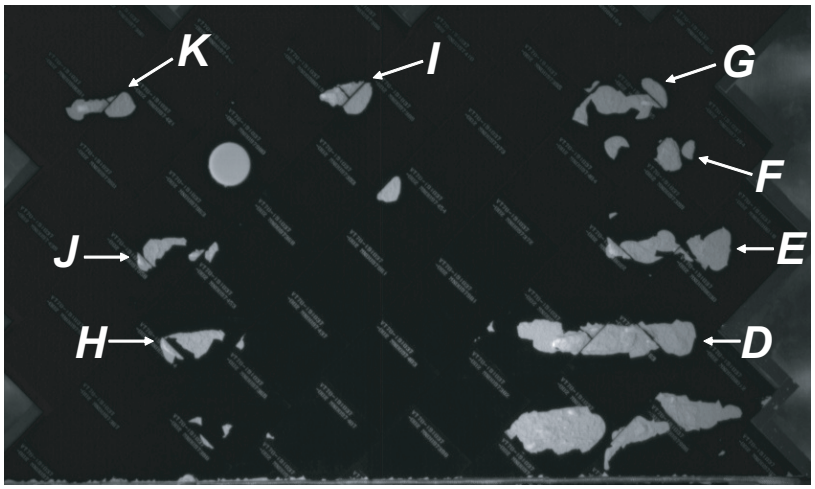

Fig. 43. Regions of Interest in Tile Damage.

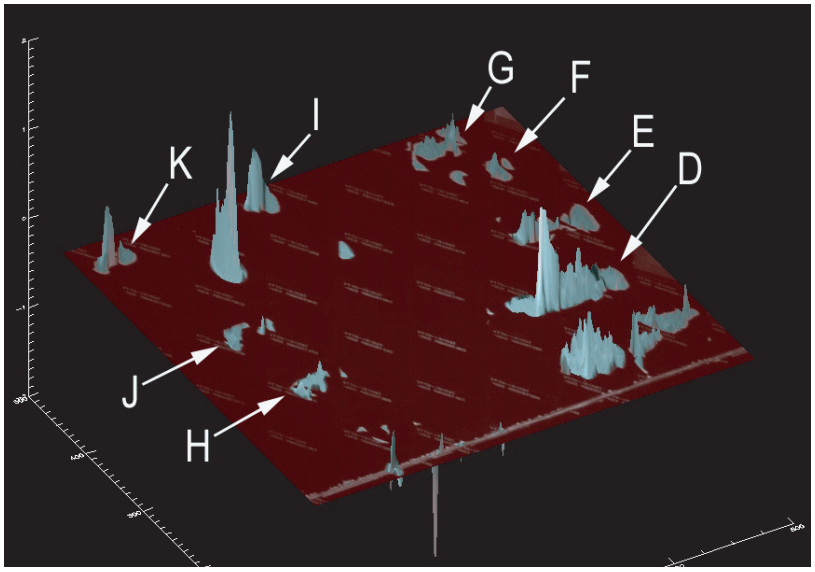

Fig. 44. Pixel Disparity for Regions of Interest.

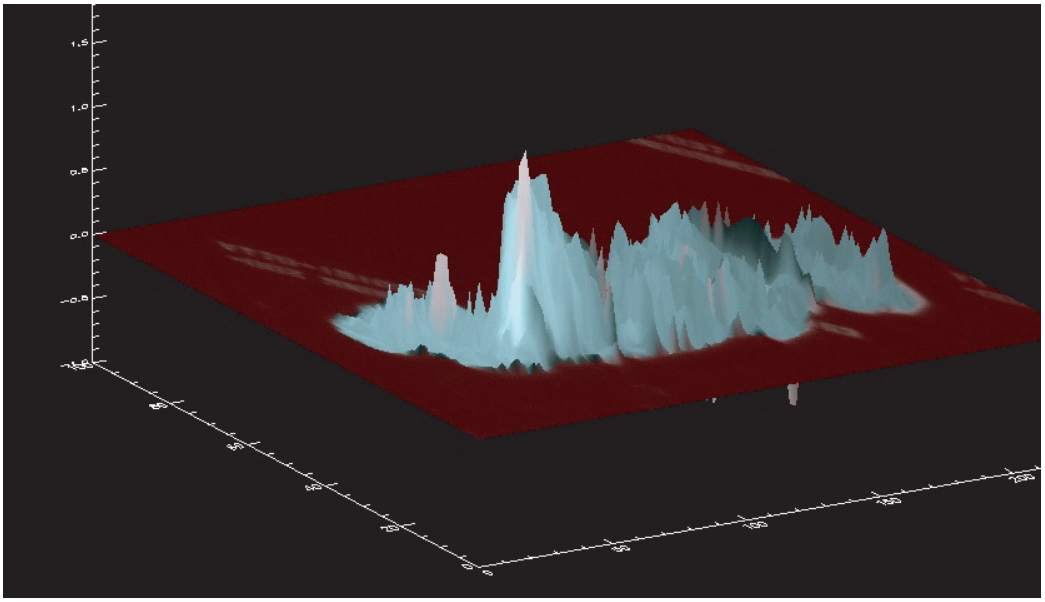

Fig. 45. Pixel Disparity for Region D.

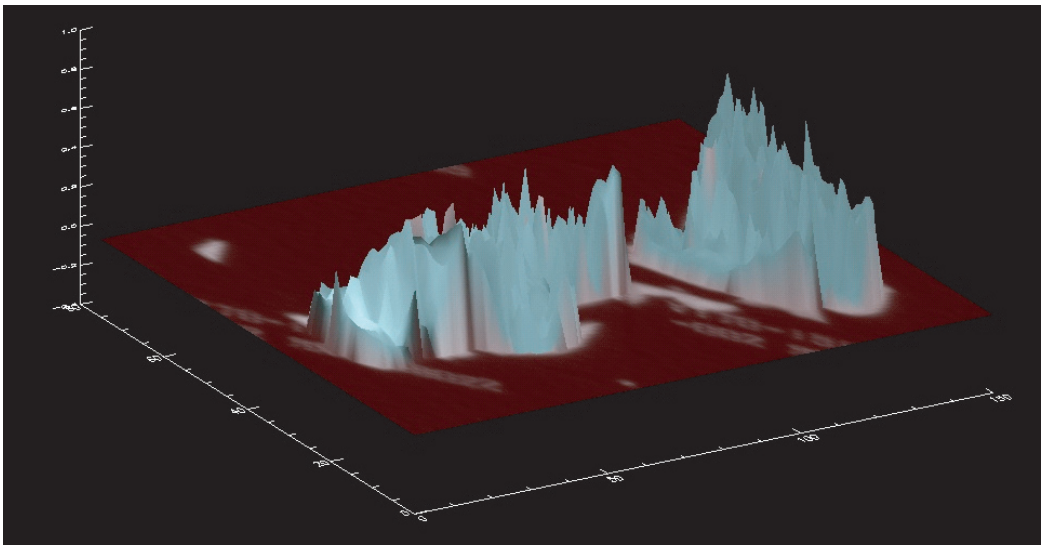

Fig. 46. Pixel Disparity for Region E.

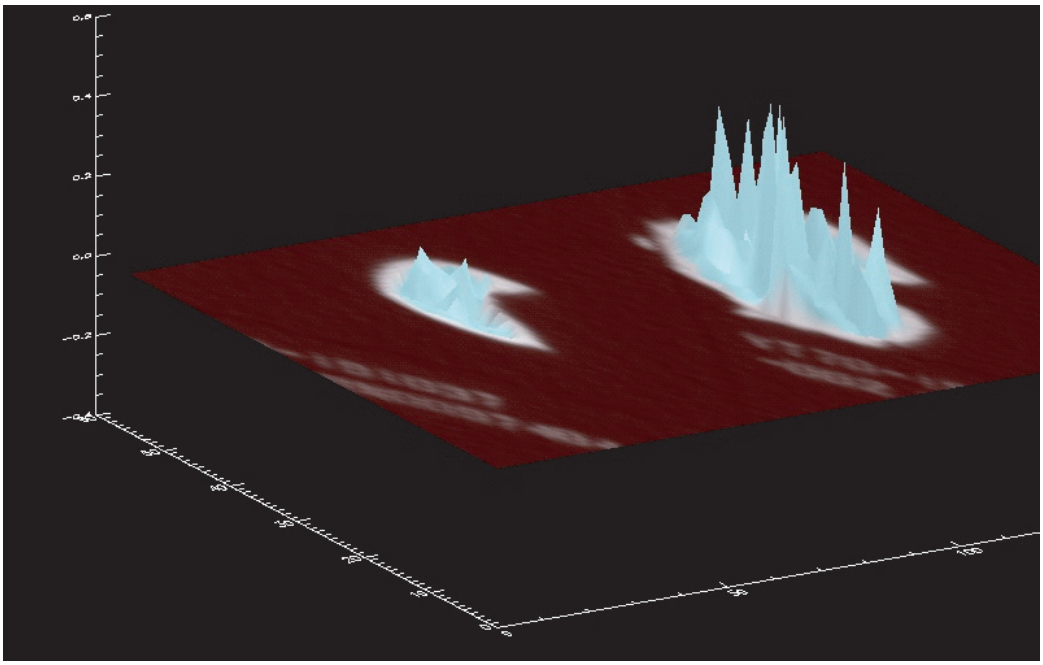

Fig. 47. Pixel Disparity for Region F.

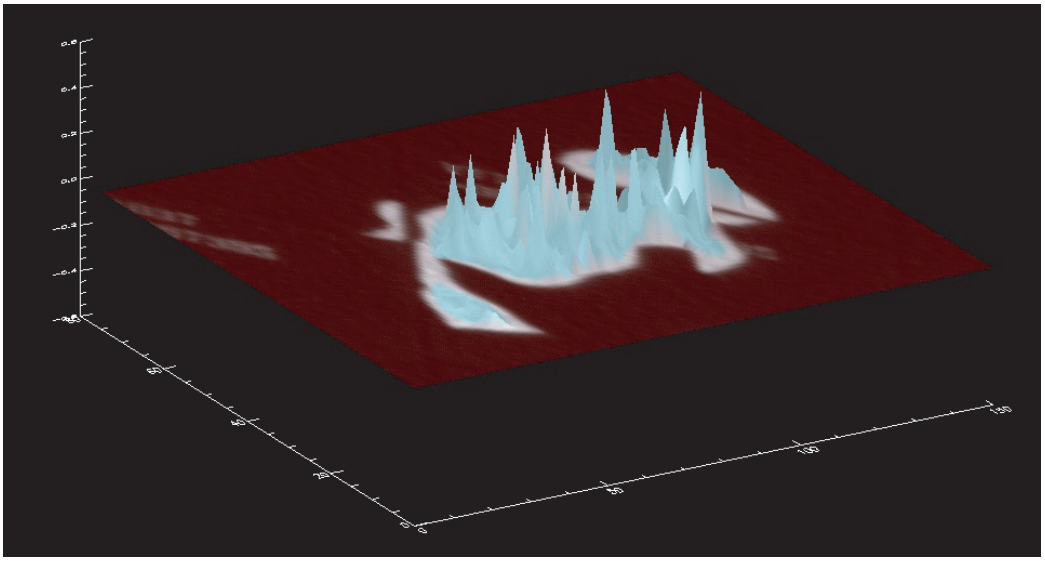

Fig. 48. Pixel Disparity for Region G.

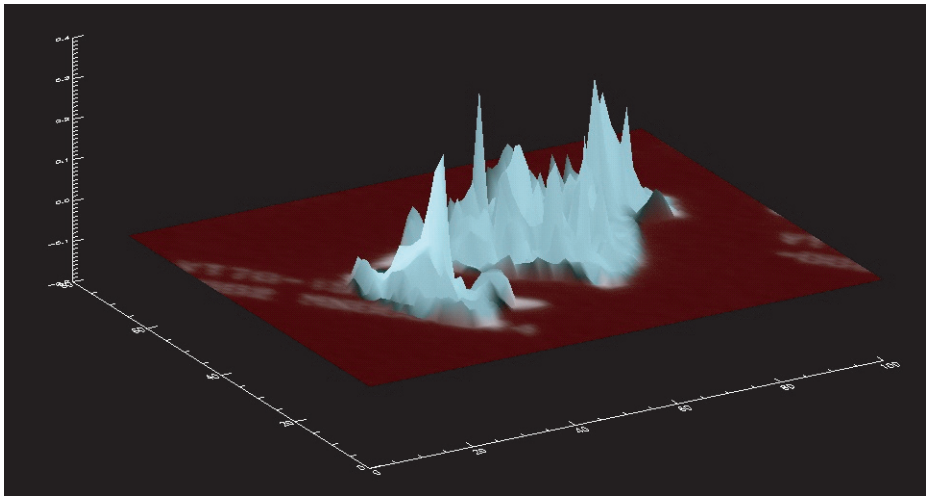

Fig. 49. Pixel Disparity for Region H.

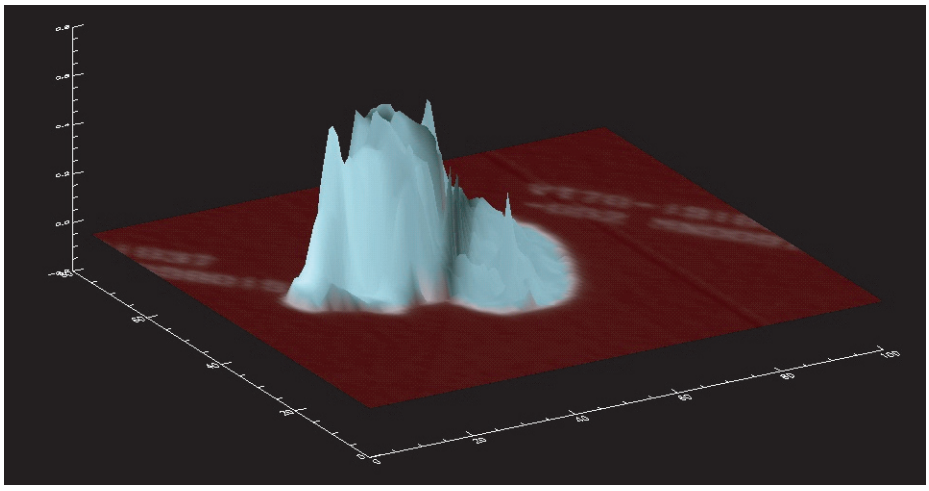

Fig. 50. Pixel Disparity for Region I.

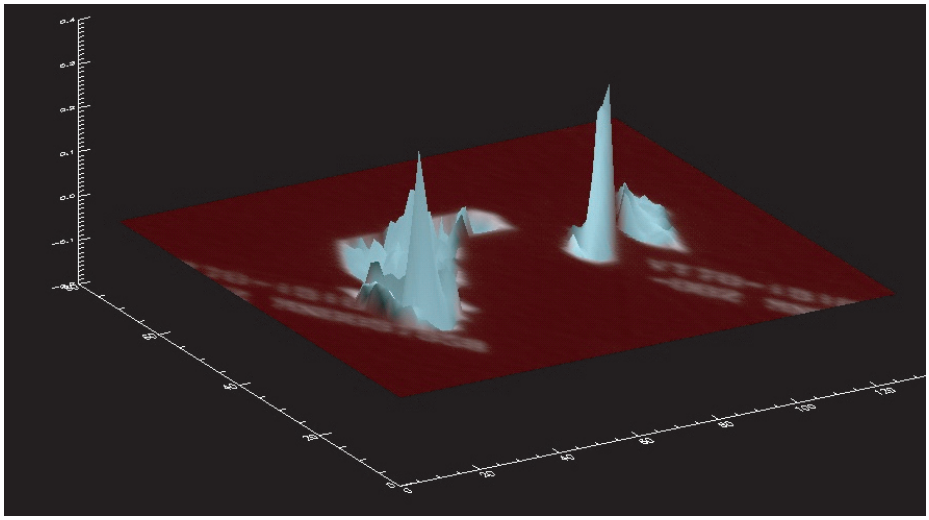

Fig. 51. Pixel Disparity for Region J.

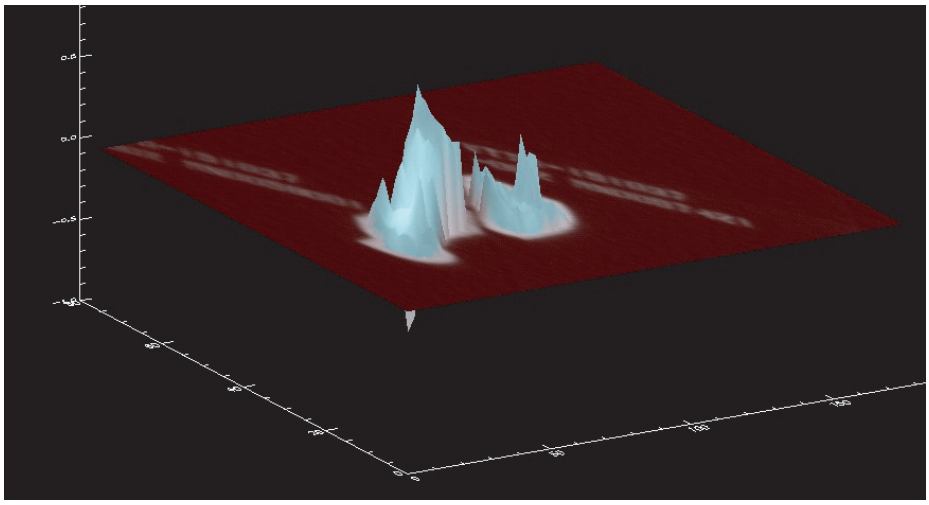

Fig. 52. Pixel Disparity for Region K.

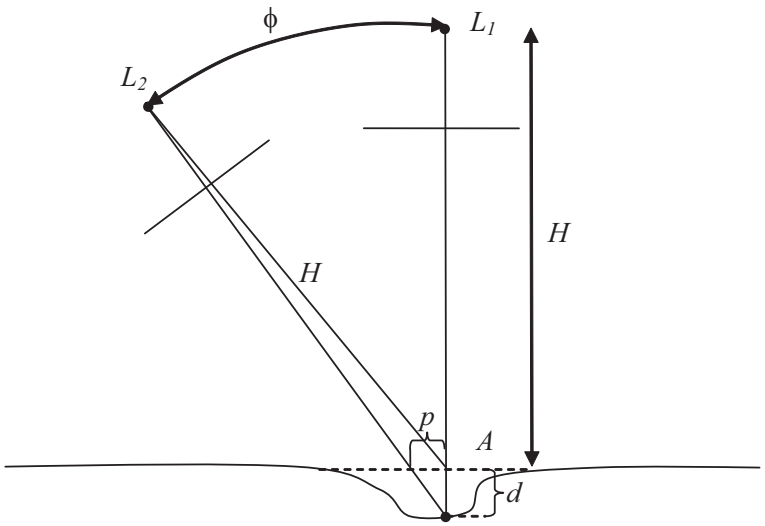

Fig. 53. Depth Determination from Disparity.

and tilt angle ϕ are shown. Both are taken at the same distance, H, from the object point, A. However, due to damage, object point A is a distance d below the surface. The shift in the expected location of object point A in the tilted image (given as the point where A would lie on the surface if were undamaged) to the actual location is given as p.

The goal is to determine d given p. The pixel shift, is converted to p simply by multiplying the pixel shift, which was found in the disparity search above, by the pixel sampling, or the distance covered by each pixel on the surface. For our test, the pixel sampling was found by measuring the metal plate used in the test, and then counting the number of pixels it took to cross it, and dividing the two numbers. In the final setup, this could be done by knowing the orientation of the orbiter and the size of certain features in the image, like the size of the square tiles.

Once the number of pixels has been converted to a distance, the damage can be

found through Equation 5.12.

$$d = \frac{p}{\tan(\phi)} \tag{5.12}$$

However, it may prove to be necessary to use two images that are both tilted. Figure 54 shows the geometry for this scenario.

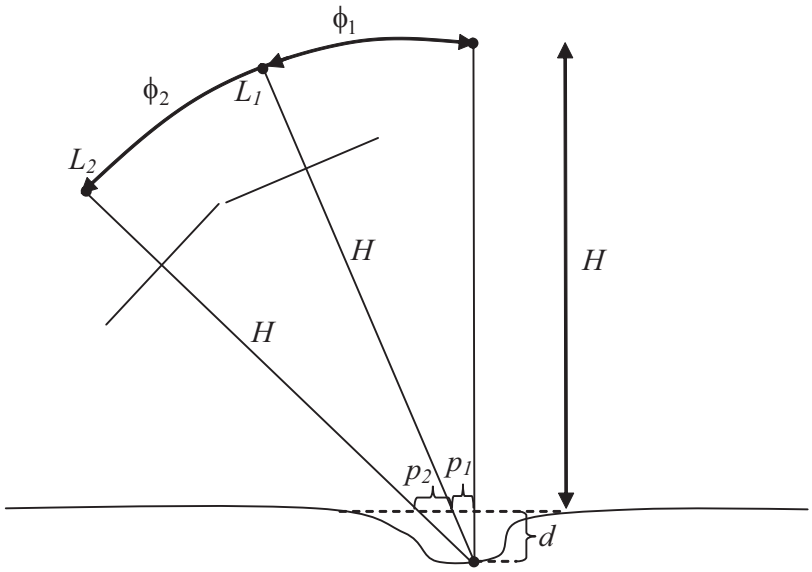

Fig. 54. Depth Determination from Disparity for Two Tilted Images.

In Figure 54, the expected location of the damage is now referenced to the image taken at station L_1. Therefore the pixel disparity that is found from the photogrammetric algorithm above, is p_2. Equations 5.13 and 5.14 show the relationships between the tilt angles and disparity shift for the two images in Figure 54.

$$tan(\phi_1) = \frac{p_1}{d} \tag{5.13}$$

$$tan(\phi_2) = \frac{p_2 + p_1}{d} \tag{5.14}$$

Subtracting Equation 5.14 from 5.13 and solving for d eliminates the unknown p_1 and

gives Equation 5.15.

$$d = \frac{p_2}{tan\,(\phi_2) - tan\,(\phi_1)} \tag{5.15}$$

One can notice that if the first angle, ϕ_1 is set to 0, Equation 5.15 reduces to Equation 5.12 as expected. This process was performed on the data given in Figures 45 - 52.

CHAPTER VI

DATA VALIDATION

The photogrammetric algorithm developed is not by any means perfect. It will have some systematic errors that arise simply from photography, and also random errors that arise from the pixel matching. It is desired, however, that even given the random errors in the photogrammetry, that the data still be valid within the range specified in Chapter I Section A.

A. Truth Comparison

A depth map of the damaged spots on the tiles was made by NASA using a laser at a very close range. This data was provided at the test to use as a truth comparison for the above processes. The truth data was very dense, and at a much higher spatial resolution than available for the imaging process. Therefore it had to be re-gridded and resampled using the techniques of Chapter III Section B2. Other processing also had to be performed on the truth data to get it into the same form to compare to the photogrammetry results. Something that can be noticed by observing Figures 45 - 52 is that the pixel-to-pixel noise ratio for the pixel disparity is very high. This is evident in the spikes and 'jumpiness' in the data. To reduce this hindrance, both the truth data and the pixel disparity data were resampled by a factor of 2. This resampling, however, still leaves enough resolution for the 0.3" lateral requirement. A comparison of the truth data to the photogrammetric results was performed on each of the areas labeled in Figure 43. However, the truth data for areas H through J were scaled incorrectly in the truth data and therefore could not be re-gridded within reasonable certainty. A comparison for areas D through G are shown in Figures 55 through 58. To show a better view of the comparison, a cross section across the

middle of the damaged areas was used in Figures 55 - 58. The drop in the edges of the truth data to around -1.5 is due to the fact that the truth data stopped at the very edge of the damage. Data that was not available in the truth data was given a large negative value for inspection.

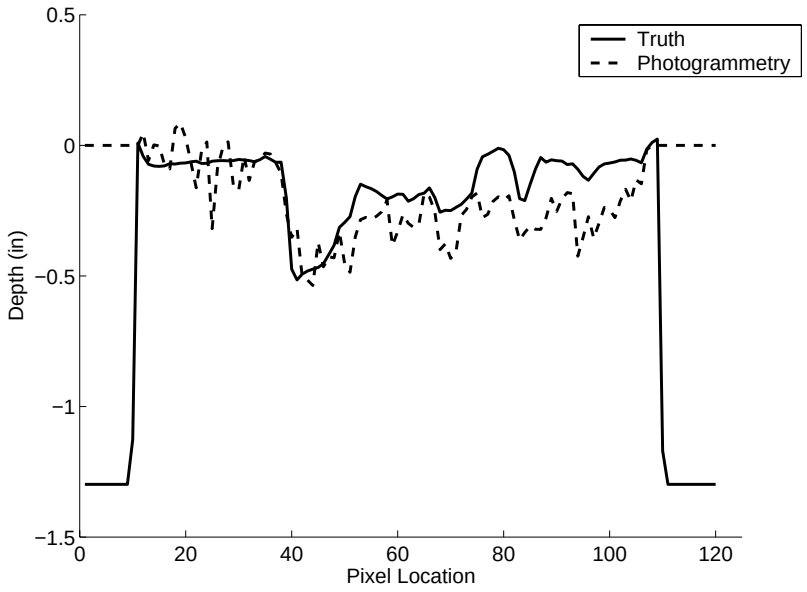

Fig. 55. Depth Comparison for Area D.

It was noted earlier that the tilt angles used in the photogrammetric algorithm were 5° and 20°. Many different angles were tried, however due to some specular regions, some of the angles were not usable. A visual comparison of each of the entire damaged areas is shown in Figure 59, where black indicates shallow, or undamaged, areas and the bright areas are deep. The damaged areas on the top half of Figure 59 are the photogrammetric data, and the damages areas on the bottom half are the truth data from NASA.

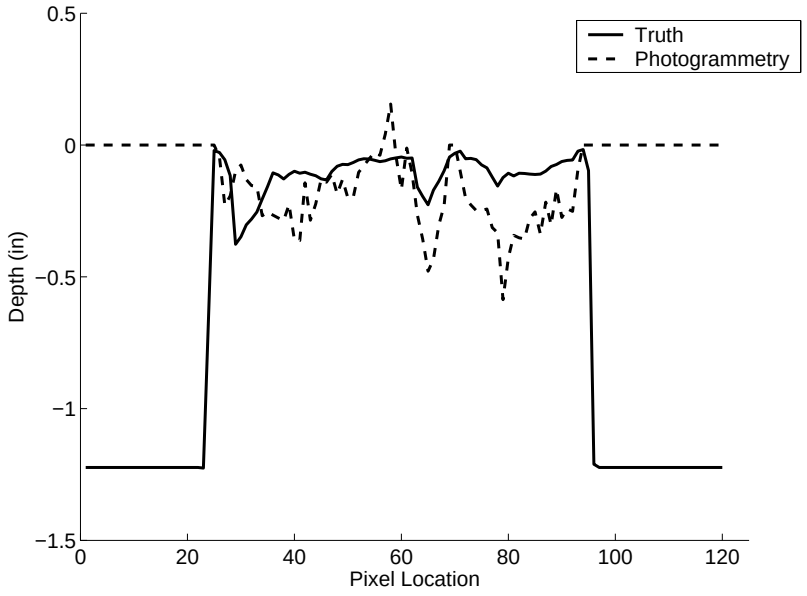

Fig. 56. Depth Comparison for Area E.

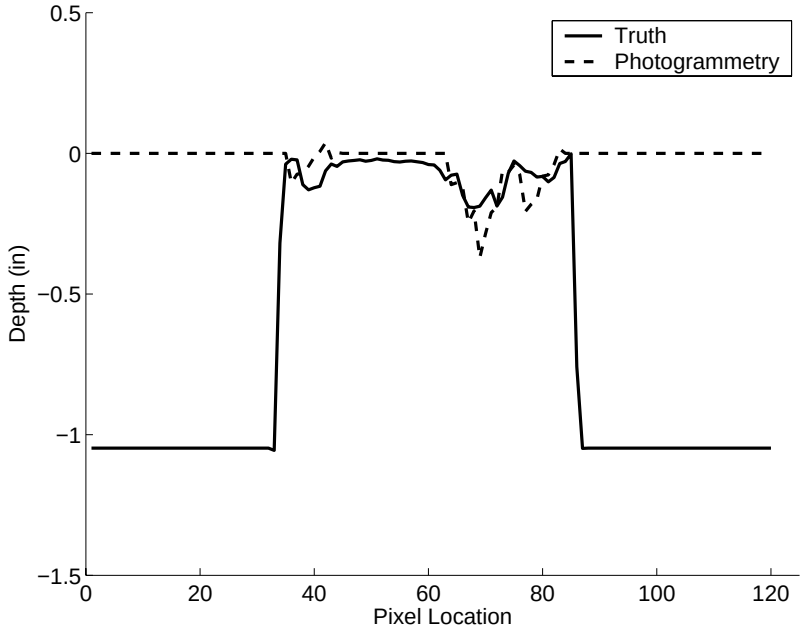

Fig. 57. Depth Comparison for Area F.

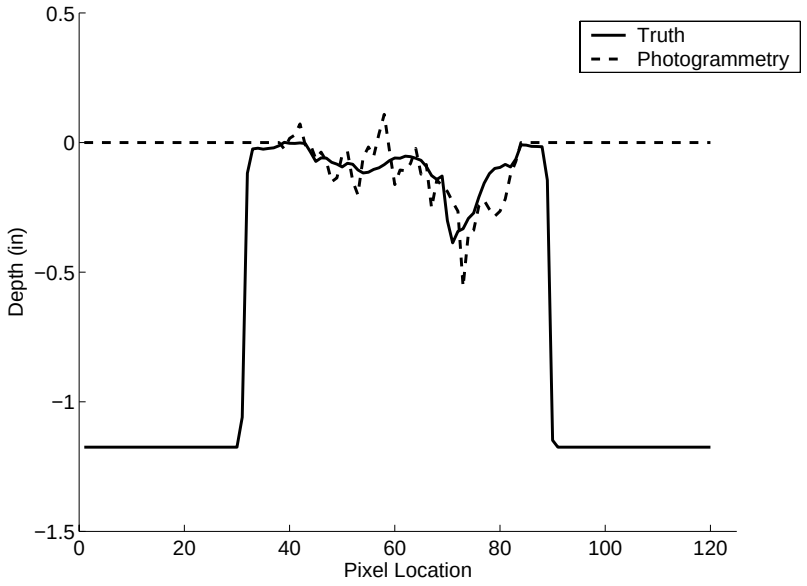

Fig. 58. Depth Comparison for Area G.

B. Error Determination

Given the requirements for inspection, it is necessary to determine the error in the photogrammetric data. Figures 60 - 63 show the error between the photogrammetric data and the truth data for the cross sections shown in Figures 55 - 58.

As can be seen from Figures 60 - 63, the photogrammetric analysis performed very well. There are, however, a couple of spots where the error between the photogrammetric analysis and the truth exceeded the error limits. However, it is to be noticed that this occurred in only a few spots and was very brief, only for a pixel or two.

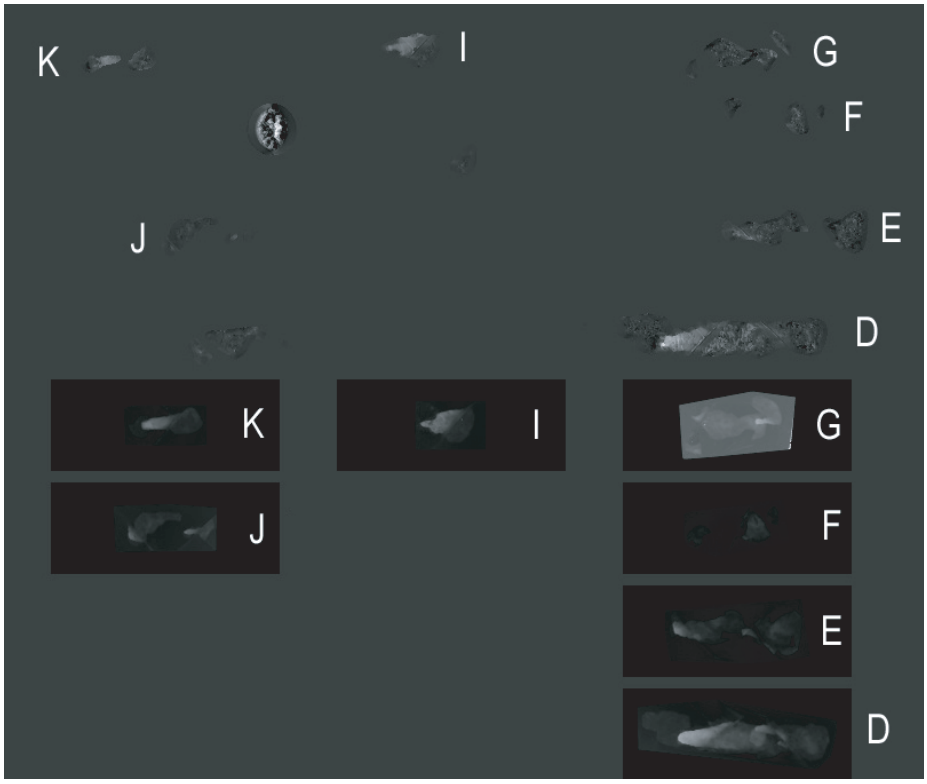

Fig. 59. Visual Comparison to Truth Data for Tile Damage.

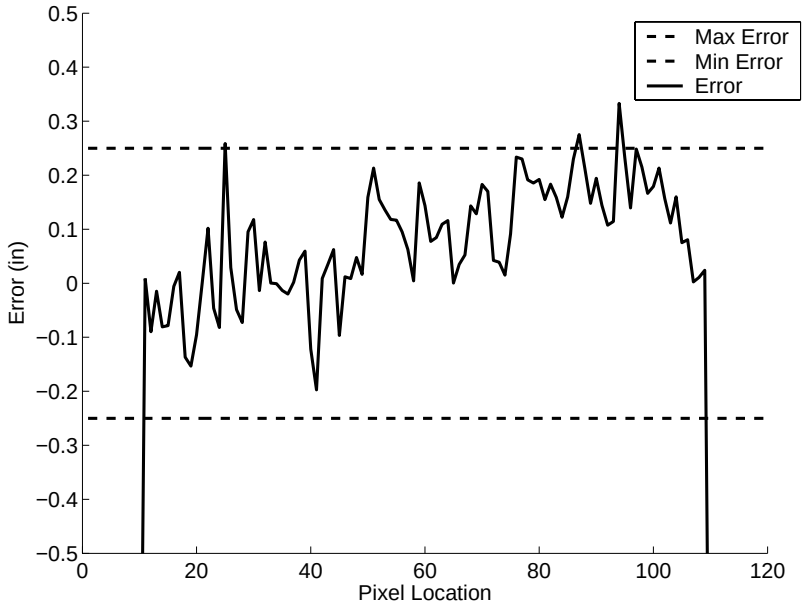

Fig. 60. Depth Error for Area D.

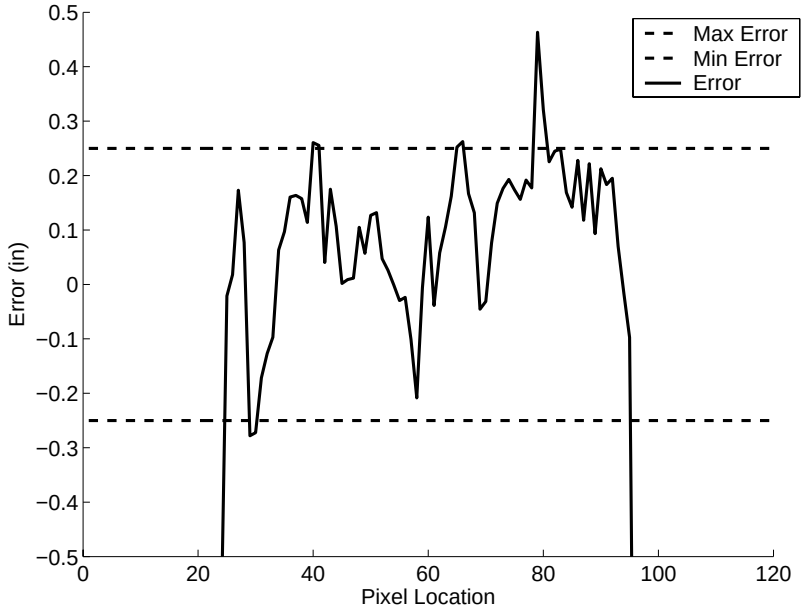

Fig. 61. Depth Error for Area E.

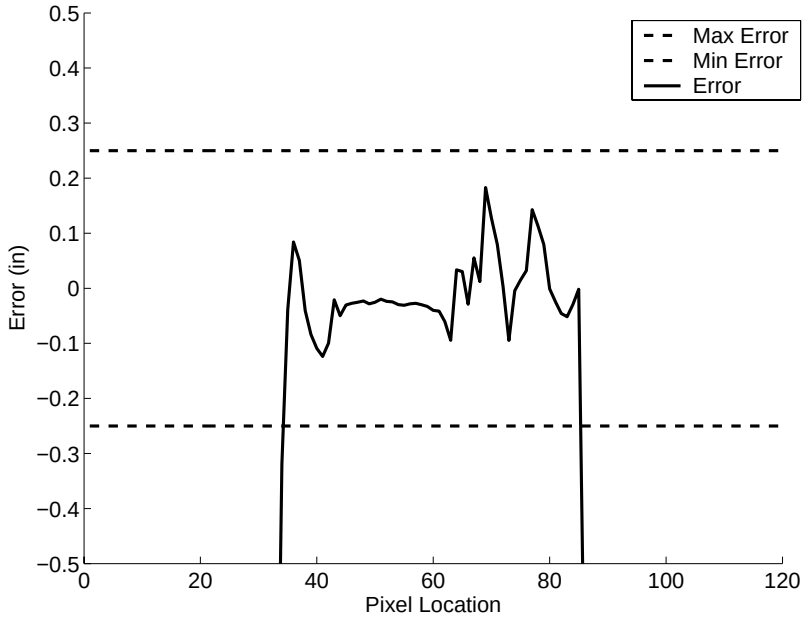

Fig. 62. Depth Error for Area F.

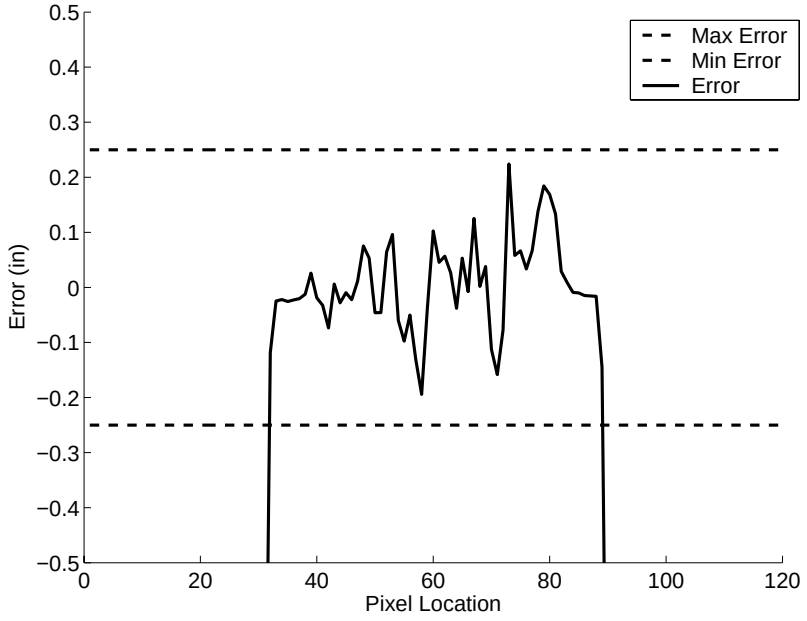

Fig. 63. Depth Error for Area G.

CHAPTER VII

CONCLUSIONS AND RECOMMENDATIONS

As was shown in Chapter VI, the photogrammetric analysis developed in Chapters IV and V performed very well. There were only a few spots in the damaged areas D through G where the error broke the 0.25" bounds. However, it is still desired that there be no places where the error is greater than 0.25". There are a few things that can be done to reduce this error.

First, there are systematic errors in the photography, or image taking process. These arise from the fact that the lens is not a perfect pin-hole. Due to lighting constraints, every camera has a finite sized aperture for which to let light through. This causes distortions in the image. These distortions were not corrected for in this process, since the final camera properties are not known. However, once the final version of the camera is completed, a calibration procedure can be performed on the camera. This will encompass taking many pictures of test patterns to determine the non-linear distortions of the camera. Using the data from such a calibration procedure, the images can be corrected back to an 'ideal' image before the photogrammetric process is performed.

There are also random effects that arise from the different feature points selected, and in the angles chosen. However, these errors will diminish once the systematic errors are reduced. Once the images are corrected for distortion, fewer feature points will be required for the registration process.

Even with the errors above, however, this process provided encouraging results for a passive Orbiter TPS inspection system. Due to the time and physical limitations from using an active system, such a passive system that can be automated on orbit and processed on the ground provides an extremely useful route.

REFERENCES

[1] C. D. Wingard, NASA Center for AeroSpace Information (CASI), "Characterization of space shuttle external tank thermal protection system (TPS) materials in support of the Columbia accident investigation," in *North American Thermal Analysis Society Conference*, Williamsburg, VA, October 2004.

[2] H. Gehmann, J. Barry, D. Deal, J. Hallock, and K. Hess, "Columbia accident investigation board, volume one," NASA Center for AeroSpace Information (CASI), Tech. Rep., August 2003, PB2003-107244,20030826.

[3] M. Hiltz, MacDonald Dettwiler Space and Advanced Robotics Ltd, "Space shuttle inspection and repair boom sensor system for return to flight," in *Proceedings of the Congress of the International Astronautical Federation*, Vancouver, Canada, October 2004.

[4] S. E. Fredrickson, S. Duran, and J. D. Mitchell, "Mini AERCam inspection robot for human space missions," in *Space 2004 Conference and Exhibit*, San Diego, CA, September 2004, AIAA 2004-5843.

[5] C. E. Harris, I. S. Raju, J. H. Stadler, R. S. Piascik, J. Kramer-White, S. G. Labbe, E. K. Ungar, H. A. Rotter, J. H. Rogers, and C. H. Null, "NESC peer-review of the flight rationale for expected debris," NASA, Tech. Rep., 2005, 2005-08-12T14:21:49Z.

[6] J. H. Chandler, "Terrain measurement using automated digital photogrammetry," in *Land Surface Evaluation for Engineering Practice, Engineering Egeology Special Publications 18, Geological Society of London*, London, 2001, pp. 13–18, iSBN 0267-9914/01.

[7] B. Lascelles, D. Favis-Mortlock, T. Parsons, and J. Boardman, "Automated digital photogrammetry: A valuable tool for small-scale geomorphological research for the non-photogrammetrist?" *Transactions in GIS*, vol. 6, pp. 5–15, 2001.

[8] B. St-Onge, J. Jumelet, M. Cobello, and C. Vega, "Measuring individual tree height using a combination of stereophotogrammetry and lidar," *Canadian Journal for Research*, vol. 34, pp. 2122–2130, 2004.

[9] D. V. Papadimitriou and T. J. Dennis, "Stereo disparity algorithm for 3d model construction," in *Fifth International Conference on Image Processing and Its Applications*, Edinburgh, UK, July 1995, pp. 178–182.

[10] P. R. Wolf and B. A. Dewitt, *Elements of Photogrammetry with Applications in GIS*. New York, NY: McGraw-Hill, 2004.

[11] E. M. Mikhail, J. S. Bethel, and J. C. McGlone, *Introduction to Modern Photogrammetry*. New York, NY: John Wiley and Sons, Inc., 2001.

[12] R. N. Bracewell, *The Fourier Transform and Its Applications*, 2nd ed. New York, NY: McGraw-Hill, 1986.

[13] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1975.

[14] M. Kasser and Y. Egels, *Digital Photogrammetry*. New York, NY: Taylor and Francis, 2002.

[15] J. L. Crassidis and J. L. Junkins, *Optimal Estimation of Dynamic Systems*. New York, NY: Chapman and Hall, 2000.

VITA

Peter Paul Gesting was born and raised in Alvin, Texas. He received a Bachelor of Science degree in Aerospace Engineering from Texas A&M University in December 2003. He entered the graduate program in Aerospace Engineering at Texas A&M University and received a Master of Science degree in Aerospace Engineering in December 2005. His research interests include aircraft and spacecraft dynamics and control. Throughout his time at Texas A&M, Paul was thoroughly involved in Habitat for Humanity and Central Baptist Church.

Paul can be reached at Texas A&M University, Department of Aerospace Engineering, TAMU 3141, College Station, TX 77843-3141. His e-mail address is gesting@tamu.edu.

Not to us, O LORD, not to us
But to Your name give glory